

**IMPLEMENTASI METODE KLASIFIKASI *SUPPORT VECTOR MACHINE* (SVM) TERHADAP PEMAKAIAN MINYAK GORENG**

**SKRIPSI**

**KEMINATAN TEKNIK KOMPUTER**

Diajukan untuk memenuhi sebagian persyaratan mencapai gelar  
Sarjana Komputer

Disusun oleh:

Muhammad Yusuf Ramadan

NIM: 135150300111032



**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2018**

## PENGESAHAN

IMPLEMENTASI METODE KLASIFIKASI *SUPPORT VECTOR MACHINE* (SVM)  
TERHADAP PEMAKAIAN MINYAK GORENG

SKRIPSI

KEMINATAN TEKNIK KOMPUTER

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh :  
Muhammad Yusuf Ramadan  
NIM: 135150300111032

Skripsi ini telah diuji dan dinyatakan lulus pada  
20 Desember 2018  
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Dahnial Syaudy, S.T., M.T., M.Sc.  
NIK. 2016078704231002

Tibyani, S.T., M.T.  
NIP. 196911011995121002

Mengetahui  
Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T., M.T., Ph.D  
NIP. 19710518 200312 1 001

A

## PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 11 Desember 2018



Muhammad Yusuf Ramadan  
NIM: 135150301111022

UNIVERSITAS



A

## KATA PENGANTAR

Puji syukur kehadirat Allah SWT, berkat rahmat dan karunia-Nya penulis dapat menyelesaikan penyusunan tugas akhir dengan baik. Shalawat serta salam semoga senantiasa terlimpahkan kepada Nabi Muhammad SAW.

Penulisan tugas akhir ini diajukan untuk memenuhi salah satu syarat untuk mendapat gelar sarjana pada Program Studi Informatika Keminatan Teknik Komputer, Fakultas Ilmu Komputer, Universitas Brawijaya Malang. Judul tugas akhir yang penulis ajukan adalah **“Implementasi metode klasifikasi *Support Vector Machine (SVM)* terhadap pemakaian minyak goreng”**. Penulis skripsi ini tidak lepas dari bimbingan serta dukungan dari berbagai pihak. Oleh sebab itu, penulis memberikan rasa hormat dan mengucapkan terima kasih kepada :

1. Keluarga tercinta yang sangat berperan penting dalam memberikan bantuan, dukungan, serta dorongan baik moral maupun moril dalam pengerjaan skripsi ini khususnya Ibu Mudrikah, Mbah Rukhaniyah, Adek Yuni dan juga Bapak Sudjari.
2. Keluarga Pondok Ath-thohiriah yang telah memberi banyak syafaat khususnya Almarhum Romo Yai Mujib Toyib, Al-fatimah.
3. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D selaku Ketua Jurusan Informatika.
4. Bapak Dahnia Syauqy, S.T., M.T., M.Sc. selaku Ketua Program Studi Teknik Komputer dan dosen pembimbing I yang telah banyak memberikan ilmu, bimbingan, nasehat, saran yang sangat baik dan luar biasa dalam penyusunan tugas akhir ini.
5. Bapak Tibyani, S.T, M.T. selaku dosen pembimbing II yang telah banyak memberikan ilmu, bimbingan dan nasehat yang sangat luar biasa dalam penyusunan tugas akhir ini.
6. Keluarga besar Flavanoid, Cecen, Odhi, Sam Jubir, Sam Bintang, Zhaes, Sam Tikum, Pak Kafi yang selama penelitian berlangsung memberikan dukungan, nasehat, bantuan dan canda tawanya yang tak terlupakan.
7. Teman seperjuangan skripsi, Brok gaming Ardiansyah, Bapak Aris, Mas ian Fhian, Kasmon Kasim, Pak yas Yasin, Chandra Gusti, Agasun Agas, Keluarga Piranha, Fikfikul Fikri, Bibol Habib, Agung, Risky Aul, Samid, dan

Keluarga besar Pondok Alam yang telah banyak mendukung dan memfasilitasi pada pelaksanaan penelitian ini.

8. Semua keluarga besar Teknik Komputer angkatan 2013 segala bantuannya sehingga terselesaikannya skripsi ini dan bantuan selama menjadi mahasiswa.
9. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat baik secara langsung maupun tidak langsung demi terselesaikannya tugas akhir ini.

Semoga jasa dan amal baik mendapatkan balasan dari Allah SWT. Ibarat tak ada gading yang tak retak, dengan segala kerendahan hati, penulis menyadari sepenuhnya bahwa skripsi ini masih belum sempurna. Oleh karena itu kritik dan saran untuk kesempurnaan skripsi ini, senantiasa penulis harapkan dari berbagai pihak.

Malang, 11 Desember 2018

Penulis  
yusuframadhan76@gmail.com





## ABSTRAK

Penggunaan minyak goreng secara berulang melebihi batas wajar dapat menimbulkan berbagai macam penyakit yang sangat berbahaya bagi kesehatan manusia seperti gagal jantung, stroke, jantung koroner, dan lain-lain. Namun, saat ini penggunaan minyak goreng secara berulang masih tinggi. Hal itu ditunjukkan oleh hasil penelitian di Kota Semarang menunjukkan 75% responden menggunakan minyak goreng yang sama untuk menggoreng dua kali sebanyak 61,2 persen, tiga kali sebanyak 19,6 persen, dan empat kali sebanyak 5,4 persen. Dari permasalahan tersebut, maka diperlukan sistem otomatisasi untuk mengklasifikasi pemakaian minyak goreng sehingga dapat digunakan untuk klasifikasi pemakaian minyak goreng yang telah digunakan beberapa kali secara akurat. Pada penelitian kali ini, parameter yang digunakan adalah warna dan tingkat kekeruhan minyak goreng. Penentuan klasifikasi minyak goreng berdasarkan warna dan tingkat kekeruhan minyak goreng diperoleh dari hasil pembacaan ADC sensor warna TCS3200 dan resistensi sensor *photodiode* oleh mikrokontroler Arduino uno dengan menggunakan metode *Support Vector Machine*, karena metode ini merupakan salah satu metode klasifikasi yang masih jarang digunakan, mudah dipahami, lebih akurat dan mempunyai kecepatan komputasi yang tinggi. Dari hasil pengujian yang dilakukan, diketahui persentase *error* pembacaan sensor warna TCS3200 adalah sebesar 3,31% dan sensor *photodiode* dapat bekerja dengan baik. Terlihat apabila minyak goreng keadaannya semakin keruh, maka nilai dari sensor *photodiode* semakin besar. Selanjutnya, pada pengujian sistem menggunakan metode *Support Vector Machine* dengan jumlah data latih sebanyak 60 data dan data uji sebanyak 13 data, diperoleh akurasi sebesar 92,3% dengan waktu komputasi sistem rata-rata selama 4384,53 ms.

Kata kunci: minyak goreng, sensor, klasifikasi, *Support Vector Machine*

## ABSTRACT

*The use of cooking oil repeatedly and beyond the normal limits (waste cooking oil) can cause variety of dangerous diseases to human health such as heart failure, stroke, coronary heart disease, and others. However, currently the use of recurrent cooking oil is still high. This is shown from the results of research in Semarang City, showing that 75% of respondents people use the same cooking oil for frying as much as 61.2 percent use it twice, 19.6 percent use it three times, and 5.4 percent use it as four time. Based on problems, it is necessary to have an automation system for classifying of the using cooking oil, so it can be used for classification of the use of cooking oil that has been used several times (waste cooking oil) accurately. In this study, the parameters used were the color and turbidity level of cooking oil. Determination of cooking oil classification is based on color and turbidity level of cooking oil was obtained from TCS3200 color sensor readings ADC and the resistance of the photodiode sensor by the Arduino uno microcontroller using the Support Vector Machine methods, because this method is one of the classification methods that are still rarely used, easy to understand, more accurate and has high computational speed. From the results of the tests performed, it is known the percentage error reading TCS3200 color sensor is 3.31% and the photodiode sensor can work well. So, if the cooking oil is more turbid, the value of the photodiode sensor is bigger. Furthermore, in testing the system using the Support Vector Machine method with the amount of training data as many as 60 data and test data as many as 13 data, obtained an accuracy of 92.3% with the average computing time for 4384.53 ms.*

*Keyword: cooking oil, sensor, classification, Support Vector Machine*

## DAFTAR ISI

PENGESAHAN .....	ii
PERNYATAAN ORISINALITAS .....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT .....	vii
DAFTAR ISI .....	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiv
BAB 1 PENDAHULUAN.....	15
1.1 Latar belakang.....	15
1.2 Rumusan masalah.....	17
1.3 Tujuan .....	17
1.4 Manfaat.....	17
1.5 Batasan masalah .....	18
1.6 Sistematika pembahasan.....	18
BAB 2 LANDASAN KEPUSTAKAAN .....	20
2.1 Tinjauan Pustaka .....	20
2.2 Dasar Teori .....	24
2.2.1 Minyak Goreng.....	24
2.2.2 Bahan yang Digoreng .....	24
2.2.2.1 Ikan Pindang.....	25
2.2.2.2 Tahu.....	26
2.2.3 Arduino Uno .....	27
2.2.4 <i>Liquid Crystal Display</i> (LCD).....	30
2.2.5 Sensor TCS3200 .....	33
2.2.6 Sensor <i>Photodiode</i> .....	37
2.2.7 Klasifikasi .....	38
2.2.8 <i>Support Vector Machine</i> .....	38
2.2.8.1 Support Vector Machine (SVM) Linier .....	39
2.2.8.2 Support Vector Machine (SVM) Non-Linier .....	40
2.2.8.3 Sequential Training .....	40



2.2.8.4 Testing SVM.....	41
2.2.9 Evaluasi.....	42
<b>BAB 3 METODOLOGI .....</b>	<b>43</b>
3.1 Studi Literatur .....	43
3.2 Analisis Kebutuhan Sistem.....	44
3.2.1 Kebutuhan Perangkat Keras.....	44
3.2.2 Kebutuhan Perangkat Lunak .....	44
3.3 Perancangan Sistem.....	44
3.4 Implementasi Sistem .....	44
3.5 Pengujian dan Analisis .....	45
3.6 Kesimpulan dan Saran .....	46
<b>BAB 4 Rekayasa kebutuhan.....</b>	<b>47</b>
4.1 Deskripsi Umum.....	47
4.2 Rekayasa Kebutuhan.....	47
4.2.1 Kebutuhan Antarmuka .....	47
4.2.2 Kebutuhan Perangkat Keras.....	48
4.2.3 Kebutuhan Perangkat Lunak .....	49
4.2.4 Kebutuhan Komunikasi .....	50
4.2.5 Kebutuhan Fungsional.....	50
4.2.5.1 Fungsi System Pendeteksi Warna .....	50
4.2.5.2 Fungsi System Pendeteksi Kekeruhan .....	50
4.2.5.3 Fungsi Klasifikasi Menggunakan Metode SVM .....	50
4.2.5.4 Fungsi Tampilkan Hasil .....	51
4.2.6 Kebutuhan Non Fungsional.....	51
4.3 Batasan Desain Sistem .....	51
<b>BAB 5 PERANCANGAN DAN IMPLEMENTASI .....</b>	<b>52</b>
5.1 Perancangan Sistem.....	52
5.1.2 Perancangan Perangkat Keras .....	53
5.1.3 Perancangan Perangkat Lunak.....	55
5.1.3.1 Perancangan Keseluruhan Sistem.....	55
5.1.3.2 Perancangan Algoritma Support Vector Machine (SVM) .	56
5.1.3.3 Perhitungan Sequential Training SVM .....	57
5.1.3.4 Perhitungan Matriks Hessien .....	58

5.1.3.5 Perhitungan Nilai $\delta\alpha_i$ .....	59
5.1.3.6 Perhitungan Nilai $\alpha_i$ .....	59
5.1.3.7 Proses Perhitungan Testing SVM .....	60
5.1.3.8 Perhitungan Manual Sequential Training SVM.....	61
5.1.3.9 Perhitungan Manual Testing SVM .....	65
5.2 Implementasi Sistem .....	68
5.2.1 Implementasi Perangkat Keras .....	68
5.2.2 Implementasi Perangkat Lunak.....	70
5.2.2.1 Pembacaan Nilai Sensor Warna TCS 3200 .....	70
5.2.2.2 Pembacaan Nilai Sensor Kekeruhan Photodiode.....	71
5.2.2.3 Pengiriman Data Menggunakan Simple Json.....	71
5.2.2.4 Algoritma Perhitungan Matriks Hessian .....	72
5.2.2.5 Algoritma Perhitungan Nilai $E_i$ .....	73
5.2.2.6 Algoritma Perhitungan Nilai $\delta\alpha_i$ .....	74
5.2.2.7 Algoritma Perhitungan Nilai $\alpha_i$ .....	74
5.2.2.8 Algoritma Perhitungan Nilai Kernel $x^+$ , Kernel $x^-$ , dan Bias .....	75
5.2.2.9 Algoritma Perhitungan Nilai Kerneltest .....	75
5.2.2.10 Algoritma Nilai Fxtest.....	76
5.2.2.11 Algoritma Tampilan Hasil Akhir pada LCD .....	76
BAB 6 Pengujian dan analisis .....	77
6.1 Pengujian Fungsional Sistem .....	77
6.1.1 Pengujian Sensor Warna TCS3200 .....	77
6.1.1.1 Tujuan.....	77
6.1.1.2 Prosedur .....	78
6.1.1.3 Hasil dan Analisis.....	79
6.1.2 Pengujian Sensor Fotodiode .....	80
6.1.2.1 Tujuan.....	80
6.1.2.2 Prosedur .....	81
6.1.2.3 Hasil dan Analisis.....	81
6.1.3 Pengujian Tampilan LCD.....	82
6.1.3.1 Tujuan.....	82
6.1.3.2 Prosedur .....	82
6.1.3.3 Hasil dan Analisis.....	83
6.2 Pengujian Akurasi <i>Support Vector Machine</i> .....	84

6.2.1 Tujuan.....	84
6.2.2 Prosedur .....	84
6.2.3 Hasil dan Analisis.....	85
6.3 Pengujian Waktu Komputasi Sistem.....	85
6.3.1 Tujuan.....	85
6.3.2 Prosedur .....	86
6.3.3 Hasil dan Analisis.....	86
BAB 7 Penutup .....	88
7.1 Kesimpulan.....	88
7.2 Saran .....	88
DAFTAR PUSTAKA.....	90
LAMPIRAN .....	92



## DAFTAR TABEL

Tabel 2.1 Penelitian Sebelumnya .....	20
Tabel 2.2 Pemilihan <i>mode photodiode</i> pada TCS3200 .....	36
Tabel 5.1 Koneksi PIN Perangkat Keras.....	54
Tabel 5.2 Data Latih .....	61
Tabel 5.3 Perhitungan Matriks <i>Hessian</i> .....	62
Tabel 5.4 Perhitungan Nilai $E_i$ pada Iterasi ke 1 .....	63
Tabel 5.5 Perhitungan Nilai $\delta\alpha_1$ pada Iterasi ke 1.....	63
Tabel 5.6 Perhitungan Nilai $\alpha_1$ pada Iterasi ke 1 .....	64
Tabel 5.7 Perhitungan nilai $E_i$ pada Iterasi ke 2 .....	64
Tabel 5.8 Perhitungan Nilai $\delta\alpha_i$ pada Iterasi ke 2 .....	64
Tabel 5.9 Perhitungan Nilai $\alpha_i$ pada Iterasi ke 2 .....	65
Tabel 5.10 Data Uji .....	65
Tabel 5.11 Nilai Matriks <i>Hessian</i> .....	66
Tabel 5.12 Nilai $X^+$ dan $X^-$ .....	66
Tabel 5.13 Perhitungan Nilai $K(x_i, x_{test1})$ dan nilai $K(x_i, x_{test2})$ .....	67
Tabel 5.14 Hasil Klasifikasi .....	68
Tabel 5.15 Spesifikasi Perangkat Keras.....	68
Tabel 5.16 Spesifikasi Perangkat Lunak .....	70
Tabel 5.17 Kode Program Sensor Warna TCS 3200 .....	70
Tabel 5.18 Kode Program Sensor <i>Photodiode</i> .....	71
Tabel 5.19 Kode Program Arduino <i>Json</i> .....	71
Tabel 5.20 Kode Program <i>Simple Json</i> .....	72
Tabel 5.21 Proses Matrik <i>Hessian</i> .....	72
Tabel 5.22 Proses Nilai $E_i$ .....	73
Tabel 5.23 Proses Nilai $\delta\alpha_i$ .....	74
Tabel 5.24 Proses Nilai $\alpha_i$ .....	74
Tabel 5.25 Proses Nilai <i>Bias</i> .....	75
Tabel 5.26 Proses Nilai <i>Kerneltest</i> .....	75
Tabel 5.27 Proses Nilai $F_{xtest}$ .....	76
Tabel 5.28 Proses Menampilkan Hasil Pada LCD .....	76
Tabel 6.1 Hasil pengujian sensor warna TCS 3200.....	79

Tabel 6.2 Pengujian Sensor <i>Photodiode</i> .....	81
Tabel 6.3 Pengujian fungsional LCD 16x2 .....	83
Tabel 6.4 Pengujian akurasi <i>system</i> .....	85
Tabel 6.5 Kode program perhitungan <i>time</i> pada <i>Python</i> .....	86
Tabel 6.6 Pengujian performa <i>system</i> .....	86





## DAFTAR GAMBAR

Gambar 2.1 Ikan Pindang .....	25
Gambar 2.2 Tahu .....	26
Gambar 2.3 Skematik Arduino Uno R3 .....	27
Gambar 2.4 Bagian <i>detail</i> Arduino Uno .....	29
Gambar 2.5 Contoh gambar LCD 16x2 .....	31
Gambar 2.6 Rangkaian dalam LCD 16x2 .....	32
Gambar 2.7 Sensor warna TCS3200 .....	34
Gambar 2.8 Skematik Sensor warna TCS3200 .....	35
Gambar 2.9 Bagian <i>source code</i> yang diubah .....	36
Gambar 2.10 Skematik <i>Photodiode</i> .....	37
Gambar 2.11 Gambar fisik LED <i>Photodiode</i> .....	38
Gambar 2.12 <i>Hyperplane</i> SVM .....	39
Gambar 3.1 Diagram alur penelitian .....	43
Gambar 5.1 Diagram blok sistem .....	52
Gambar 5.2 Desain Perancangan Sistem .....	53
Gambar 5.3 Skematik Perancangan Perangkat Keras .....	54
Gambar 5.4 Perancangan Pengambilan Data Latih .....	56
Gambar 5.5 Proses <i>Support Vector Machine</i> .....	56
Gambar 5.6 Perhitungan <i>Sequential Training</i> SVM .....	57
Gambar 5.7 Perhitungan Matriks <i>Hessian</i> .....	58
Gambar 5.8 Perhitungan Nilai $\delta\alpha_i$ .....	59
Gambar 5.9 Perhitungan Nilai $\alpha_i$ .....	60
Gambar 5.10 Perhitungan <i>Testing</i> SVM .....	60
Gambar 5.11 <i>Prototype</i> Alat Pendeteksi Kelayakan Minyak Goreng .....	69
Gambar 5.12 Tempat Peletakan Gelas Uji Minyak Goreng .....	69
Gambar 6.1 Grafik Perbandingan Nilai kekeruhan .....	82

## BAB 1 PENDAHULUAN

### 1.1 Latar belakang

*Cooking oil* atau yang disebut juga sebagai minyak goreng, merupakan bahan pangan yang sering digunakan dan dikonsumsi dalam kehidupan sehari-hari. Minyak goreng memiliki komponen utama yang terkandung di dalamnya berupa kalori yang tidak terpakai dan disimpan pada sel lemak dengan bahan nabati tanpa ada campuran kimia. Termasuk *hydrogen*, yang dalam pembuatannya telah melalui beberapa proses pemurnian untuk kemudian dapat digunakan untuk menggoreng (BSNI, 2013). Salah satu kebutuhan pokok ini perlu dijaga kualitas dan mutunya supaya seluruh masyarakat terhindar dari berbagai macam masalah kesehatan yang disebabkan oleh keteledoran dalam menggunakan minyak goreng yang sudah tidak layak. Karena pengguna minyak goreng curah sendiri didominasi oleh kalangan Pedagang Kreatif Lapangan (PKL) yang diperoleh dari restoran cepat saji maupun hotel, untuk mengurangi beban produksi. Padahal minyak terdapat batasan-batasan tertentu dalam penggunaannya karena semakin banyak minyak tersebut digunakan maka kadar minyak jenuh akan dapat mengakibatkan efek yang buruk bagi kesehatan (Aminah, 2010).

Terdapat berbagai parameter untuk mengetahui minyak goreng tersebut masih memiliki kualitas yang baik atau tidak yaitu dengan mengamati sifat fisik dan kimia yang terdapat pada minyak goreng. Perubahan warna pada minyak goreng, bau yang ditimbulkan, titik didih yang mulai berubah, dan kekeruhan yang dapat terlihat oleh mata telanjang merupakan parameter yang berupa sifat fisik dari minyak goreng. Sedangkan *hydrogen* untuk mengubah minyak nabati menjadi olesan (*Hidrogenasi*), mampu menghasilkan lemak dan *gliserol* (*Hidrolisis*), dan kerusakan mutu minyak goreng disebabkan oleh *Oksigen* (*Oksidasi*) merupakan parameter berupa sifat kimia dari minyak tersebut (AminahS, 2010). Standar mutu pada minyak goreng berpengaruh besar untuk menentukan komposisi bahwa minyak goreng tersebut termasuk dalam kelas baik atau tidak. Kandungan air yang terdapat pada minyak, perubahan warna minyak, perubahan bilangan peroksida, kandungan asam lemak bebas dan kotoran yang terkumpul didalam minyak merupakan faktor penentu standar mutu minyak goreng. Proses penyulingan dengan memperhatikan syarat mutu minyak goreng yang masih baik merupakan salah satu solusi untuk mengatasi masalah produksi pangan khususnya pedagang kaki 5. Namun alangkah baiknya pedagang menggunakan minyak goreng yang sudah berstandart SNI dalam proses produksinya. Karena minyak goreng sulingan tidak memiliki standart tertentu dalam penentuan kualitas minyak yang baik untuk digunakan kembali. Kandungan minyak jenuh yang terdapat pada minyak goreng sulingan bisa mengakibatkan orang yang terlalu sering mengkonsumsinya menjadi *obesitas* dan bisa juga bisa mengakibatkan *stroke* dan gangguan penyakit berbahaya yang lainnya (Malkan, 2015).

Pada penelitian pertama, parameter yang digunakan berupa warna dan tingkat kekeruhan minyak goreng. Untuk mendapatkan hasil klasifikasi dari minyak

goreng. Peneliti menggunakan sensor warna yaitu sensor TCS3200 untuk mendeteksi perubahan warna minyak goreng. Kemudian sensor *photodiode* untuk mengetahui tingkat kekeruhan pada minyak goreng. Dari input berupa 2 sensor tersebut kemudian diproses kedalam *mikrokontroler* dengan menggunakan algoritma *Bayes*. Pengujian dari sensor warna TCS3200 menghasilkan nilai pembacaan dengan presentase *error* 1,9% dan sensor *photodiode* dapat bekerja dengan baik. Terlihat apabila minyak goreng keadaannya semakin keruh, maka nilai dari sensor fotodioda semakin kecil. Kemudian pada pengujian system dengan menggunakan metode *Bayes*, didapatkan hasil akurasi system dengan nilai presentase 71,42. Hasil tersebut didapatkan dari 65 data latih dan 35 data uji dengan rata-rata waktu komputasi sistem (*millis*) selama 13,144 ms (Marofi, 2017).

Penelitian kedua dengan menggunakan parameter yang sama dilakukan penelitian untuk menentukan klasifikasi frekuensi penggorengan pada minyak goreng dengan mendeteksi warna dari R(*Red*), G(*Green*), B(*Blue*) diperoleh dari hasil pembacaan kamera *raspberry pi* dan untuk kekeruhan diperoleh dari pembacaan LDR(*Light Emitting Diode*) oleh *Raspberry Pi 3* dengan menggunakan metode KNN(*K-Nearest Neighbor*). Dari hasil pengujian yang dilakukan diketahui persentase akurasi dari pembacaan R(*Red*), G(*Green*), B(*Blue*) pada kamera *raspberry pi* dengan sensor TCS3200 adalah sebesar R = 89,964%, G = 86,139%, B = 82,293%. Pada pengujian sistem yang menggunakan metode KNN(*K-Nearest Neighbor*) dengan jumlah data latih 72 data dan data uji 30 data, diperoleh akurasi K=1, K=3, K=5 sebesar 73,33% dengan waktu komputasi sistem rata – rata selama 3,9 ms (Putri, 2018).

Berdasarkan penelitian sebelumnya, akan dilakukan perbandingan tingkat akurasi dan kinerja system dari metode yang sebelumnya di pakai yaitu metode *Bayes* dengan metode SVM (*Support Vector Machine*). SVM sendiri merupakan metode *learning machine* yang bertujuan mendapatkan batas atau jarak terbaik yang dapat memisahkan antara 2 kelas, juga dapat disebut dengan istilah *hyperplane* (Satriyo, 2003). *Artificial Neural Network* masih satu kelas dengan SVM dalam proses penyelesaian permasalahan dan termasuk dalam kelas *supervised learning*. Sebagai pendukung dalam proses penelitian ditambahkan sensor warna sebagai pembaca kondisi minyak tersebut. Modul sensor warna dan cahaya juga digunakan sebagai pemroses pada prototipe. Sensor warna TCS3200 bekerja dengan memantulkan cahaya dari LED yang kemudian pantulan dari objek yang berada didepannya diterima oleh IC TCS3200 yang didalamnya terdapat 2 sensor *photodiode* dan juga converter dengan memanfaatkan intensitas warna yang tampak pada minyak goreng untuk mem-filter warna merah, hijau, biru dan clear. Osilator yang terdapat pada modul menghasilkan nilai pulsa yang memiliki frekuensi sebanding dengan nilai perubahan warna yang dideteksi pada minyak goreng.

Pada penelitian ini, metode SVM dipilih untuk dapat menentukan fungsi *hyperplane* terbaik dari 2 kelas yang berbeda dengan 3 kali proses pengujian pada

setiap object. Penggunaan metode *learning machine* SVM diharapkan menghasilkan performa yang akurat dalam memprediksi 2 kelas yang berbeda.

## 1.2 Rumusan masalah

Pada perumusan masalah yang disebutkan, dapat diambil kesimpulan untuk mendapatkan rumusan masalah .

1. Bagaimana merancang system yang dapat menentukan kualitas minyak goreng dengan memanfaatkan sensor TCS3200 sebagai pembaca warna berupa nilai  $R$ ,  $G$ ,  $B$  dan sensor *photodiode* berupa nilai  $P$  sebagai pembaca tingkat kekeruhan minyak goreng?
2. Berapa tingkat akurasi metode *SVM* yang diterapkan pada penelitian ini untuk menentukan kualitas dalam pemakaian minyak goreng?
3. Berapa waktu komputasi pemrosesan sistem klasifikasi penggunaan minyak pada penelitian dengan metode *SVM* ?

## 1.3 Tujuan

Berdasarkan permasalahan yang disebutkan diatas, penelitian ini bertujuan untuk mendapatkan hasil sebagai berikut.

1. Mendapatkan nilai pembacaan sensor warna yang baik dengan melakukan kalibrasi yang akurat pada sensor TCS3200 dalam menentukan nilai  $R$ ,  $G$ ,  $B$  dan menempatkan sensor *photodiode* pada tempat yang memungkinkan mendapatkan hasil pembacaan sensor dengan baik.
2. Mendapatkan nilai akurasi yang baik dengan menggunakan metode *SVM* untuk menentukan kualitas penggunaan minyak goreng.
3. Mengukur waktu komputasi pemrosesan sistem klasifikasi penggunaan minyak pada penelitian dengan metode *SVM*.

## 1.4 Manfaat

Dari permasalahan yang diangkat dalam penelitian, diharapkan mendapatkan manfaat sebagai berikut

1. Dapat memberikan solusi kepada dinas kesehatan maupun pengusaha makanan khususnya yang menggunakan minyak goreng sebagai bahan utama dalam produksi untuk menentukan kualitas dan kelayakan minyak goreng dengan mudah.
2. Memberikan edukasi pada masyarakat dalam proses penggunaan minyak goreng tidak melebihi batas maksimal agar tetap terjaga kualitas minyak goreng tersebut dan kesehatan pada konsumennya.
3. Memberikan referensi untuk penetian lainnya dalam pengembangan metode *SVM* dalam bidang yang sama maupun bidang yang lain.

## 1.5 Batasan masalah

Beberapa batasan yang terdapat pada penelitian akan dijelaskan seperti berikut.

1. Untuk menentukan kelayakan minyak goreng dilakukan pembacaan warna minyak goreng dengan memanfaatkan sensor warna TCS3200 dan kekeruhan minyak goreng dengan memanfaatkan sensor *photodiode*.
2. Bahan penggorengan yang digunakan untuk menentukan kelayakan minyak goreng menggunakan ikan dan tahu.
3. Pemilihan minyak goreng untuk proses penelitian adalah minyak goreng curah dan Bimoli.
4. Dalam penelitian ini, menggunakan 2 bahasa pemrograman yaitu *Python* sebagai pengolah data dalam bentuk *matrix* dan Arduino IDE sebagai pengolah *input* dan menampilkan *output*.
5. Pada proses pembuatan data latih dilakukan dengan cara dilakukan penggoreng dengan suhu 100<sup>0</sup> C, 2l minyak goreng, dan dengan estimasi waktu selama 5 menit.
  1. Penggorengan tahu 1 kali.
  2. Penggorengan tahu 2 kali.
  3. Penggorengan tahu 3 kali.
  4. Penggorengan ikan 1 kali.
  5. Penggorengan ikan 2 kali.
  6. Penggorengan ikan 3 kali.

## 1.6 Sistematika pembahasan

Merupakan gambaran alur system yang akan digunakan pada penelitian untuk memudahkan dalam memahami alur penelitian yang akan dilaksanakan, antara lain seperti berikut:

### BAB I Pendahuluan

Membahas penelitian mengenai latar belakang, kemudian didapatkan rumusan masalah dari permasalahan, mendapatkan tujuan dari penelitian, manfaat untuk orang lain, serta batasan masalah yang terkait dengan Implementasi Metode Klasifikasi *Support Vector Machine* (SVM) Terhadap Pemakaian Minyak Goreng.

### BAB II Landasan Kepustakaan

Membahas penelitian-penelitian terdahulu yang dapat menunjang penelitian "Implementasi Metode Klasifikasi *Support Vector Machine* (SVM) Terhadap Pemakaian Minyak Goreng" berupa dasar teori, objek, sensor, mikrokontroler, dan algoritma yang digunakan yaitu SVM.



### **BAB III Metodologi Penelitian**

Dalam metodologi pada penelitian mendiskripsikan suatu metode dalam langkah-langkah membangun suatu system untuk implementasi metode klasifikasi *Support Vector Machine* (SVM) terhadap pemakaian minyak goreng.

### **BAB IV Rekayasa Kebutuhan**

Pada bagian rekayasa kebutuhan, menjelaskan tentang kebutuhan perangkat keras berupa laptop dan Arduino Uno sebagai sarana dari perangkat lunak yang digunakan, yaitu *Python* dan juga Arduino IDE. Dan juga *prototype* sebagai alat penguji dari minyak yang sudah di goreng menggunakan bahan ikan maupun non ikan.

### **BAB V Perancangan dan Implementasi**

Membahas tentang hal-hal yang berkaitan dengan perancangan alat yang akan dibuat untuk dapat mendeteksi kelayakan minyak goreng dengan menggunakan metode SVM. Kemudian mengimplementasikan perancangan system tersebut.

### **BAB VI Pengujian dan Analisis**

Menunjukkan hasil proses pengujian dari tiap-tiap sensor dan komponen yang digunakan oleh peneliti. Kemudian melakukan pengujian system dengan menggunakan sebuah metode. Selanjutnya menghitung komputasi performa system.

### **BAB VII Penutup**

Pada bagian penutup memuat hasil yang didapat dari proses pengerjaan dan juga pengujian dari penelitian yang telah dilaksanakan, kemudian didapatkan saran dari kekurangan yang terdapat pada penelitian untuk dilakukan pengembangan lebih lanjut.

## BAB 2 LANDASAN KEPUSTAKAAN

### 2.1 Tinjauan Pustaka

Dalam tinjauan pustak ini membahas tentang penelitian-penelitian yang sudah ada sebelumnya sebagai bahan acuan yang berkaitan dengan penelitian yang saat ini diusulkan.

**Tabel 2.1 Penelitian Sebelumnya**

No.	Judul Paper	Penjelasan	Perbedaan Penelitian
1.	Uji Kualitas Minyak Goreng Pada Para Penjual Goreng Di Lingkungan Kampus Universitas Jember (Prasetyawan, 2007).	Penjual gorengan biasanya menggunakan minyak goreng curah. Sementara pengukuran kualitas minyak goreng dengan parameter yang berbeda mengalami perubahan	Pada jurnal tersebut dalam mengambil sample perubahan fluktuasi pada minyak goreng dengan mengkalibrasi kedalam mikrokontroler untuk mendapatkan hasil yang sesuai dengan data yang didapat.
2.	Bilangan Peroksida Minyak Goreng Curah Dan Sifat Organoleptik Tempe Pada Pengulangan Penggorengan (AminahS, 2010)	Semakin banyak Pengulangan penggorengan, peningkatan peroksida mengakibatkan nilai yang tidak baik pada minyak goreng.	Dalam jurnal menjelaskan tentang penggunaan suhu yang tinggi pada proses penggorengan mengakibatkan penurunan kualitas minyak. Sehingga batas penggunaan minyak goreng dalam proses penggorengan dibatasi sebanyak tiga kali.
3.	Rancang Bangun Mesin Pertamina Digital Untuk Minyak Goreng (Rohmah, 2017)	Rancang bangun mesin pertamini digital minyak goreng untuk mempermudah pekerjaan manusia. Dengan menggunakan sensor water flow sebagai alat ukur cairan keluarnya. Water Pump untuk proses pemompaan cairan dan	Rancang bangun system klasifikasi minyak goreng dengan menggunakan Arduino uno sebagai mikrokontroler pengkalibrasi dari sensor yang digunakan.

		arduino nano sebagai pusat kendali dari rancang bangun tersebut.	
4.	Praktek Penggorengan Dan Mutu Minyak Goreng Sisa Pada Rumah Tangga Di Rt V Rw III Kedungmundu Tembalang Semarang (Aminah, 2010)	75 % responden yang menggunakan minyak goreng curah. Dalam proses penggorengan menggunakan minyak goreng bekas penggorengan sebelumnya yang ditambahkan dengan minyak goreng baru untuk menggoreng beberapa jenis makanan.	Pada jurnal dalam penerapan terhadap alat dapat langsung membuktikan tanpa harus bertanya pada responden tentang minyak goreng yang digunakan tersebut masih layak digunakan atau tidak dengan menguji kekeruhan dan warna minyak tersebut.
5.	Sistem Penyeleksi Dan Pengelompokan Produk Berdasarkan Warna Berbasis Programmable Logic Controller (Yang, 2015)	Mengelompokkan produk sesuai dengan warna diperlukan sebuah sistem yang dapat mengelompokkan berdasarkan warna produk. Sistem dirancang menggunakan PLC sebagai unit kontrol sistem pneumatik, yang berfungsi menggerakkan sensor warna serta microcontroller yang berfungsi mengolah data keluaran sensor warna. Warna dikelompokkan menjadi tiga yaitu biru, merah, dan hijau.	Pada jurnal tersebut menggunakan sensor TCS3200 sebagai pengelompokan produk berdasarkan warna. Sedangkan pada penelitian yang akan dilakukan sensor TCS3200 digunakan untuk membaca perubahan warna pada minyak goreng
6.	Rancang Bangun Sistem Deteksi Minyak Goreng Yang telah Dipakai Menggoreng Daging Babi	Pengujian minyak goreng yang mengandung Babi menggunakan sensor photodiode. Sensor photodiode	Jurnal tersebut membahas tentang sensor pada penelitian yang kemudian digunakan untuk mendeteksi penggunaan

	Menggunakan LED dan Photodiode (Arifin, 2015).	merupakan sensor yang peka terhadap cahaya.	minyak goreng dengan parameter kekeruhan dan warna minyak goreng.
7.	Kualitas Minyak Goreng dan Produk Gorengan Selama Penggorengan di Rumah Tangga Indonesia (Malkan, 2015)	Standart mutu untuk mengetahui kualitas minyak goreng tersebut baik atau tidak dengan memperhatikan beberapa faktor diantaranya kadar air, kotoran dan warna minyak serta kadar asam lemak bebas. Proses penyulingan dengan memperhatikan syarat mutu minyak goreng yang masih baik merupakan salah satu solusi untuk mengatasi masalah produksi pangan khususnya pedagang makanan. Kandungan minyak jenuh yang terdapat pada minyak goreng sulingan bisa mengakibatkan <i>obesitas</i> dan gangguan penyakit berbahaya lainnya jika terlalu sering di konsumsi.	Pada paper, menggunakan tahu sebagai bahan uji kualitas minyak goreng. Sedangkan pada penelitian yang akan dilakukan menggunakan bahan uji tahu dan ikan asin untuk mengukur parameter penggunaan minyak goreng.
8.	Desain dan Realisasi Alat Deteksi Kecerahan Minyak Goreng Kelapa Sawit Menggunakan Sensor <i>Optocoupler</i> Berbasis Mikrokontroler	Dalam penelitian tersebut menggunakan sensor LDR sebagai pendeteksi cahaya dari LED merah, hijau, dan biru. Nantinya LED tersebut akan mempresentasikan suatu warna pokok	Pada paper, menggunakan sensor LDR sebagai pendeteksi cahaya dari LED merah, hijau, dan biru. Nantinya LED tersebut akan mempresentasikan suatu warna pokok <i>Red, Green, dan Blue</i> (RGB) yang

	ATMega8535 (Yuliana, 2014)	Red, Green, dan Blue (RGB) yang diperlukan untuk membentuk suatu warna.	diperlukan mengetahui warna dari minyak goreng, sedangkan dalam penelitian ini digunakan sensor TCS3200 yang digunakan sebagai pendeteksi warna RGB pada minyak goreng.
9.	Rancang Bangun Sistem Klasifikasi Frekuensi Penggunaan Minyak Goreng Dengan Menggunakan Metode Bayes (Marofi, 2017)	Sistem deteksi tersebut digunakan sensor TCS3200 serta Photodiode yang berfungsi sebagai pendeteksi seberapa sering minyak goreng digunakan untuk melakukan penggorengan ikan dan selain ikan.	Pada jurnal yang digunakan sebagai rujukan penelitian ini, sensor TCS3200 dan photodiode mempunyai fungsi untuk mendeteksi warna pada minyak goreng. Sementara itu pada penelitian yang akan dilakukan photodiode digunakan untuk mendeteksi tingkat kejernihan minyak goreng.
10.	Penerapan Metode Support Vector Machine pada Sistem Deteksi Intrusi secara Real-time (Jacobus, 2013)	Metode SVM digunakan dalam pembagian jalur lalu lintas jaringan menjadi 3 kelas.	Dalam jurnal yang dijadikan rujukan pada penelitian ini digunakan SVM yang berfungsi sebagai langkah dalam menentukan penjurusan. Namun pada penelitian yang akan dilakukan SVM akan digunakan sebagai langkah menentukan klasifikasi penggunaan minyak.
11	Uji Kelayakan Minyak Goreng Curah dan Kemasan yang Digunakan Menggoreng Secara Berulang (Nora, 2016)	Penelitian dengan menggunakan minyak goreng dalam kemasan dan minyak goreng curah yang digunakan dalam proses penggorengan hingga kedua kalinya, karena pada penggorengan ketiga	Proses penggorengan dilakukan sebanyak 3 kali dengan menggunakan objek yang berbeda sebagai parameter pengujian yaitu berupa ikan dan non ikan.



		kalinya angka peroksida dan kadar asam lemak bebas sudah melampaui ambang batas aman yang ditetapkan SNI 01-3741-2013. (Nasional, 2013).	
--	--	--	--

Pada tabel 2.1 menunjukkan penelitian-penelitian terdahulu yang sudah dilaksanakan untuk kemudian diterapkan pada penelitian yang akan dilakukan yaitu mengimplementasi metode klasifikasi *Support Vector Machine* (SVM) terhadap pemakaian minyak goreng,

## 2.2 Dasar Teori

Pada bab ini menjelaskan tentang berbagai referensi tentang komponen-komponen serta teknologi terkini yang digunakan pada penelitian untuk memberikan informasi dan spesifikasi pada komponen, teknologi dan metode yang digunakan.

### 2.2.1 Minyak Goreng

*Cooking oil* atau minyak goreng suatu zat yang mengandung kalori dan lemak yang dibutuhkan oleh tubuh dalam proses pertumbuhan. Merupakan sumber energy yang baik selain karbohidrat dan protein yang dibutuhkan oleh tubuh. Asam linoleate, arakidonat, dan lenoleat merupakan minyak nabati yang bermanfaat untuk mencegah penyempitan pada pembuluh darah yang disebabkan oleh tingkat kolesterol yang tinggi. Selain bermanfaat bagi kesehatan tubuh, minyak goreng juga dapat melarutkan beberapa vitamin seperti vitamin D, E, K, dan A yang ada di dalam tubuh kita (Ketaren, 2012).

Kualitas yang ada pada minyak tergantung dari titik didihnya, dimana dilakukan pemanasan hingga membentuk batas akreolin pada minyak. Jika melebihi dari batas akreolin akan berakibat pada rasa gatal yang terjadi pada tenggorokan. Maka dari itu mutu minyak goreng bisa dikatakan baik apabila sewaktu proses penggunaannya menggunakan titik didih yang tinggi namun dengan batasan tertentu. Dari hasil penggorengan dengan menggunakan titik didih yang sesuai dengan anjuran penggunaan tersebut akan menghasilkan lemak dan gliserol yang bermanfaat bagi tubuh yang disebut dengan *hidrolisis*. Oleh sebab itu dalam proses penggunaannya, minyak goreng harus digunakan sesuai dengan aturan dan ketentuan yang dianjurkan dalam proses penggunaannya (Winarno, 2012).

### 2.2.2 Bahan yang Digoreng

Bahan penggorengan yang digunakan pada penelitian kali ini berupa ikan pindang dan non ikan yaitu tahu. Peneliti menggunakan bahan pindang karena

pindang memiliki kadar asam lemak bebas tinggi setelah digoreng, yaitu sebesar 8,335% dan hal tersebut dapat mempengaruhi atau bahkan merusak kualitas minyak goreng. Sebaliknya, non ikan (tahu) memiliki kandungan asam lemak bebas rendah, yaitu sebesar 0,470% setelah digoreng. Dari 2 bahan tersebut mempunyai perbedaan sifat fisik minyak setelah dilakukan penggorengan dengan hasil yang cukup berbeda. Dari hasil tersebut, peneliti menggunakan bahan ikan dan non ikan untuk penelitian ini dengan tujuan supaya dapat membedakan sifat fisik minyak goreng tersebut dengan mendeteksi perubahan warna dan juga perubahan tingkat kekeruhannya. Berikut ini merupakan penjelasan dari objek yang digunakan untuk proses penelitian minyak goreng.

### 2.2.2.1 Ikan Pindang

Ikan pindang yang dijual di pasar merupakan ikan dengan hasil olahan dengan cara pemasakan dan kemudian dilakukan penggaraman. Memiliki cita rasa, keawetan dan tekstur yang begitu khas sesuai dengan kadar garam, lama perebusan dan jenis ikan yang didapatkan. Ikan selar, tembang, kembung, lemuru, selar, layang, tuna dan tongkol merupakan jenis ikan yang biasanya diolah dengan proses pemindangan. Ikan yang masih segar biasanya dipilih sebagai bahan baku untuk proses pembuatan ikan pindang. Selain dari bentuk fisiknya yang masih terlihat bagus. Ikan pindang yang sudah terkontaminasi dengan lingkungan luar maupun dalam kondisi sudah tidak segar akan mempengaruhi dalam proses penggorengan karena daging ikan akan mudah hancur dan tercampur dengan minyak yang mengakibatkan kandungan minyak menjadi cepat jenuh. Oleh karena itu peneliti lebih memilih ikan pindang yang segar pada proses penelitiannya. Ikan pindang juga sangat penting bagi proses pertumbuhan karena kaya akan protein yang dibutuhkan oleh tubuh (JTP, 2016). Berikut ini merupakan penampang dari ikan pindang yang ditunjukkan pada Gambar 2.1.



**Gambar 2.1 Ikan Pindang**  
Sumber : sipas.bekasikota.go.id

### 2.2.2.2 Tahu

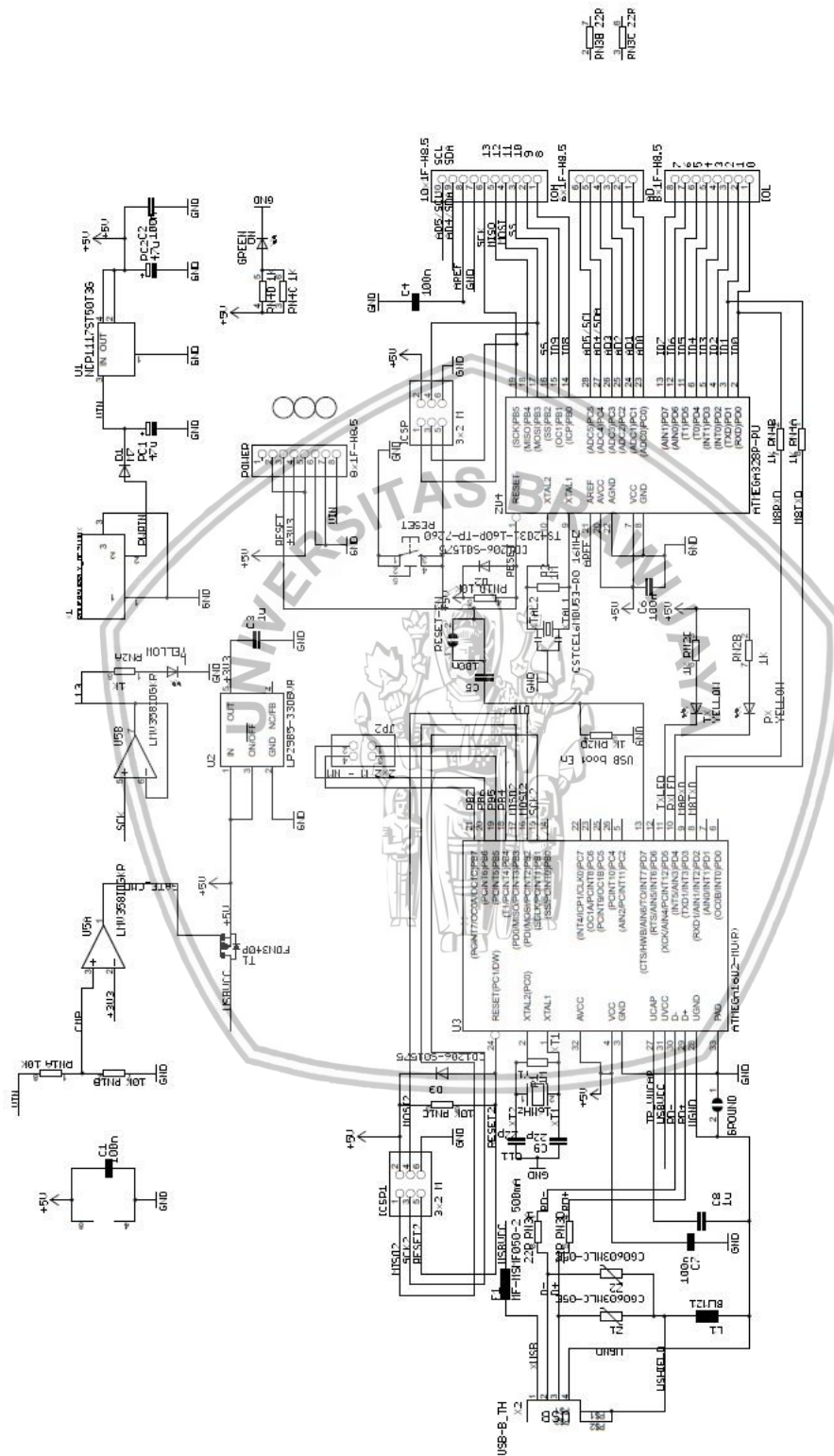
Salah satu lauk yang biasa disajikan di meja makan adalah tahu. Kedelai merupakan bahan utama dalam pembuatan tahu. Memiliki strukstur yang lunak berupa padatan. Cara pembuatan tahu dengan mengolah kedelai yang kemudian diendapkan untuk kemudian diambil endapannya berupa tahu itu sendiri yang memiliki kandungan protein yang banyak. Protein tersebut berasal dari kedelai yang memiliki sumber protein nabati dari jenis biji-bijian dan kacang-kacangan yang hampir menyerupai protein pada hewan. Kandungan asam amino pada tahu sangat baik dalam proses perkembangan pada tubuh. (Isyuniarto, 2014). Seperti yang ditunjukkan oleh Gambar 2.2 berikut ini.



**Gambar 2.2 Tahu**

Sumber : pasartemanggung.com

## 2.2.3 Arduino Uno



Gambar 2.3 Skematik Arduino Uno R3

Sumber : pratito-zicho.blogspot.co.id



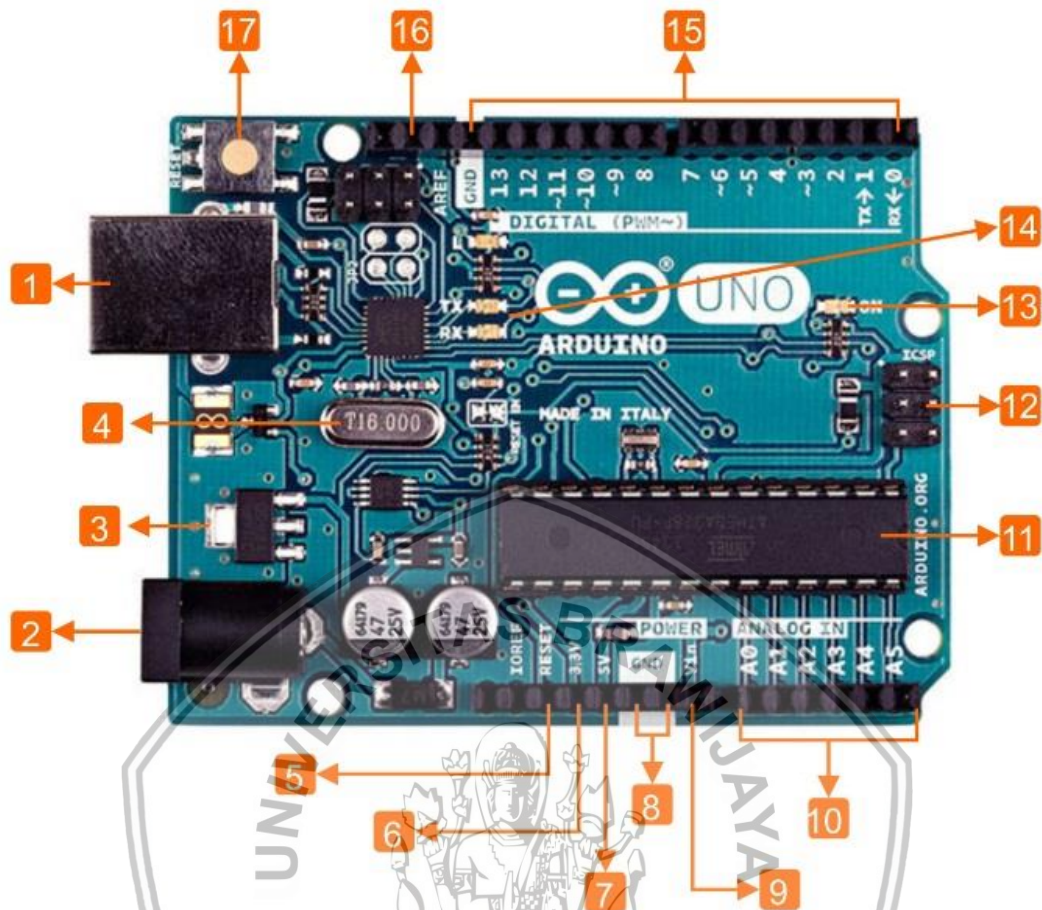
Arduino Uno mempunyai pin I/O sebanyak 14 digital pin, 6 pin untuk *analog*, 6 pin lainnya untuk *output PWM* kemudian terdapat konektor *USB*, kabel *jack*, *crystal* sebagai clock sebesar 16 MHz, tombol *reset* dan juga *header ICSP*. Penjelasan pada Gambar 2.3 merupakan berbagai komponen yang diperlukan untuk mendukung sebuah rangkaian mikrokontroler.

Arduino Uno REV 3 atau disebut juga R3 merupakan sebuah mikrokontroler yang dalam bahasa Italia diartikan kata Uno adalah Satu. *Software* yang digunakan pada pemrograman Arduino adalah *Arduino IDE* (Banzi, 2005). *Software* tersebut bertujuan untuk memudahkan dalam meng-*upload* sebuah *code* kedalam *chip* ATmega328 yang dapat dioperasikan pada *Mac*, *Linux* maupun *Windows OS*. Berikut ini merupakan komponen dalam dari Arduino REV 3.

1. mikrokontroler: ATmega328P
2. Sumber tegangan: 5 Volt
3. Sumber tegangan pada kabel *jack*: 7 Volt – 12 Volt
4. Sumber tegangan terbatas pada kabel *jack*: 6 Volt – 20 Volt
5. *Input* dan *Output* pin sebanyak 14 buah, dengan 6 pin *PWM* diantaranya
6. Pin analog sebanyak 6 buah pin
7. Tegangan DC per pin *Input*, *Output* sebesar 20 mA
8. Tegangan arus dc pada pin 3.3 Volt sebesar 50 mA
9. *Flash memory* yang digunakan untuk *bootloader* sebesar 0.5 KB – 32 KB
10. *Static Random Access Memory* sebesar 2 KB
11. *Electrically Erasable Programmable Read Only Memory* sebesar 1 KB
12. Kecepatan proses *Clock* sebesar 16 Mhz
13. Memiliki berat 25 gram

Arduino Uno dilengkapi dengan *polyfuse* yang memiliki fungsi untuk mereset mikrokontroler pada *port USB* laptop/computer dan sebagai *port* pelindung arus lebih. Meskipun *port* pelindung sudah banyak terdapat pada komputer model baru, Arduino uno sendiri juga memiliki *sikring* sebagai lapisan pelindung yang dapat membuat pengguna dengan aman menghubungkan mikrokontroler pada komputer. Cara kerja sirkuit proteksi yaitu dengan menyambungkan dan memutuskan kembali arus yang lebih dari 500 mA sampai kembali ke batasan aman.





**Gambar 2.4 Bagian detail Arduino Uno**

Sumber: ecadio.com

Penampang gambar 2.4 diatas merupakan bentuk fisik dari Arduino uno Rev 3 dengan petunjuk angka pada tiap-tiap komponen yang bertujuan untuk menjelaskan tujuan dan fungsi dari komponen tersebut.

Berikut ini merupakan penjelasan dari gambar diatas :

1. *Power USB*, berfungsi sebagai penghubung antara komputer dengan Arduino, maupun Arduino dengan *power supply*.
2. *Power Jack*, sumber tegangan pada Arduino dengan komponen tipe *jack* dengan arus 5 sampai 12 Volt.
3. *Voltage Regulator*, IC berfungsi menstabilkan listrik dari power tipe *jack* menuju ke pin 5v Arduino dengan aman.
4. *Crystal Oscillator*, sebagai fungsi clock pada Arduino untuk menentukan proses mana yang akan dijalankan terlebih dahulu dengan besar daya 12 MHz.
5. *Reset*, pin dengan perintah mengulang program yang ditunjukkan pada gambar 2.4 dengan angka 17.
6. 3.3V, sumber tegangan yang terdapat pada Arduino dengan daya 3.3 Volt.

7. 5V, sumber tegangan yang terdapat pada Arduino dengan daya 5 Volt.
8. *Ground*, berfungsi sebagai akhir dari jalannya tegangan yang masuk.
9. *Vin*, berfungsi sebagai pemasok tegangan dari Arduino sebesar 5 Volt..
10. *Pin Analog*, terdapat enam analog pin pada Arduino yaitu A0-A5. Digunakan untuk membaca analog sensor berupa sensor suhu, jarak, dll. Kemudian mengubah nilai analog yang di dpan menjadi digital. *Serial Clock Line* (SCL) berfungsi untuk menghantar *clock* sinyal. Selanjutnya *Serial Data* (SDA) digunakan untuk mentransmisikan data dari Arduino ke computer.
11. *IC Mikrokontroler*, berfungsi sebagai otak Arduino yang mengatur jalannya semua proses dari pin digital yang ditunjukkan angka 15 dan analog pada angka 10.
12. *ICSP pin*, terdapat 6 pin didalam ICSP. Yang termasuk golongan SPI yaitu MOSI, berfungsi menerima data atau disebut juga dengan *download*. MISO, berfungsi untuk mengirim data pada Arduino atau disebut *downloader*. SCK, berfungsi untuk jalur *sinkronisasi* dalam mengatasi kesalahan sewaktu berkomunikasi, dll.
13. *Power Led Indicator*, indikator untuk mengetahui bahwa Arduino sudah bisa dijalankan, apabila indikator ini tidak menyala berarti terdapat kesalahan pada Arduino.
14. *LED RX dan TX*, berfungsi sebagai jalur komunikasi penerima RX dan pengirim data TX pada mikrokontroler.
15. *Digital Pins I/O*, berfungsi sebagai saklar pada pin yang digunakan, dengan logika analog 0 dan 1 dengan *range* 0-255 pada PWM .
16. *AREF*, berfungsi sebagai sumber tegangan 0 sampai 5 Volt yang dapat diatur lewat IC mikrokontroler.

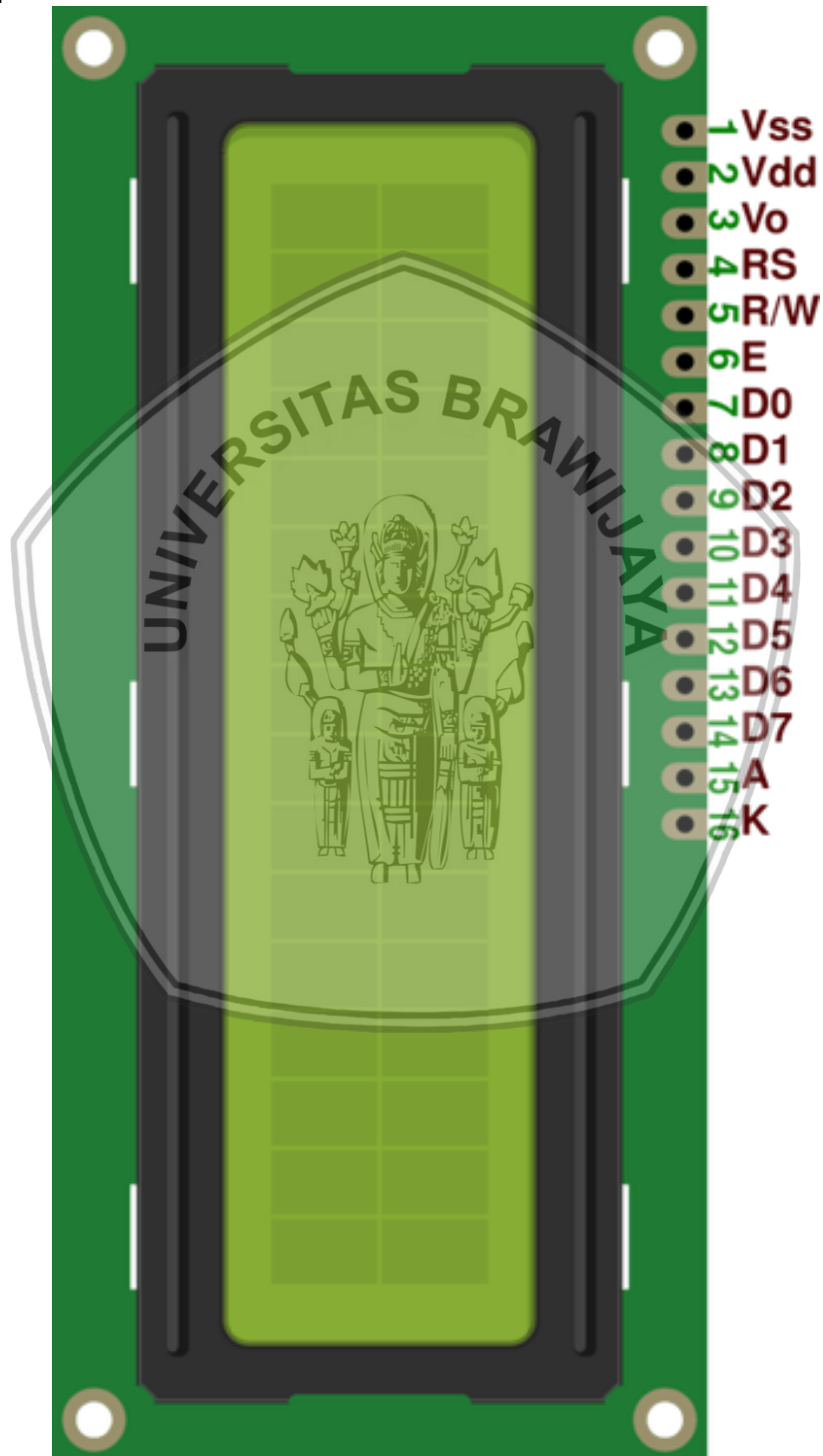
Pada Arduino uno rev 3, LED RX dan TX akan berkedip-kedip ketika ada komunikasi yang ditransmisikan pada *chip* USB menuju Serial dengan kabel USB pada laptop atau komputer. *Chip* ATmega328 memiliki fungsi *support* pada SPI dan I2C (TWI). *Wire Library* berfungsi memudahkan penggunaan bus I2c pada *Software* Arduino IDE.

#### 2.2.4 Liquid Crystal Display (LCD)

LCD merupakan suatu komponen yang memiliki kegunaan sebagai tampilan tulisan, angka, dan karakter dari suatu data. CMOS *logic* berfungsi mentransmisikan cahaya dari *black-lit* atau memantulkan cahaya yang ada sekelilingnya pada *front-lit*. Seperti ditunjukkan pada Gambar 2.5.

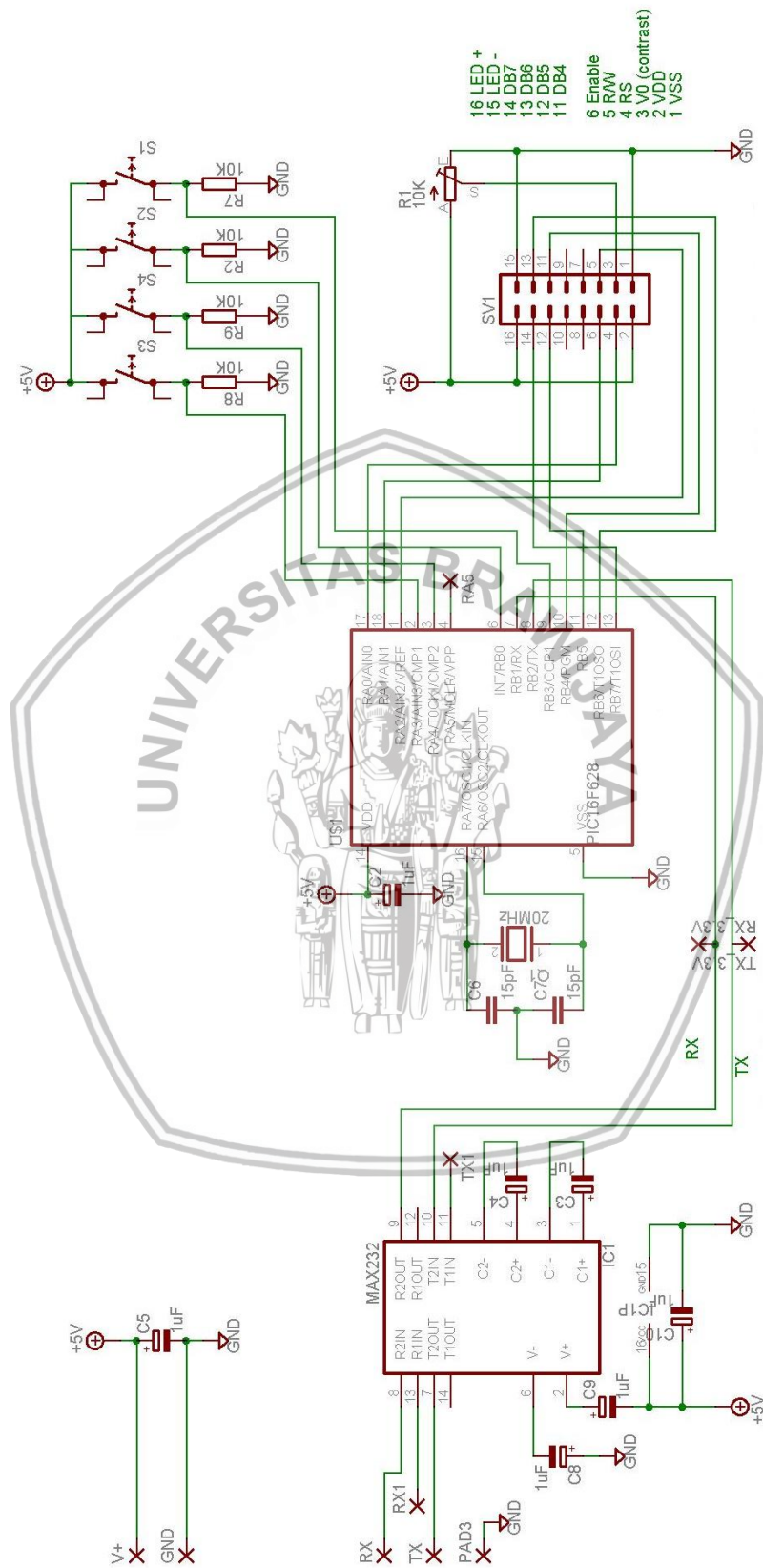
LCD merupakan campuran dari lapisan kaca bening campuran organik dengan elektoda transparan pada oksida indium berupa tampilan elektroda pada kaca bening dan juga *seven-segment*. Saat elektroda aktif dengan mendapatkan tegangan, silindris dan molekul organik yang panjang, kemudian segmen

menyesuaikan diri dengan elektroda yang didapat. *Sandwich* pada LCD mempunyai *polarizer* cahaya horisontal belakang dan *polarizer* cahaya vertikal depan yang diikuti dengan lapisan *reflektor*. Karakter data yang ditampilkan berasal dari segmen yang diaktifkan, sehingga terlihat gelap dan membentuk karakter. Selanjutnya cahaya yang dipantulkan tidak dapat melewati molekul-molekul pada LCD.



Gambar 2.5 Contoh gambar LCD 16x2

Sumber: elektronika-dasar.web.id



**Gambar 2.6 Rangkaian dalam LCD 16x2**



Gambar 2.6 menunjukkan rangkaian skematik didalam komponen LCD. *Module Liquid Cristal Display* (LCD) terdapat mikrokontroler yang memiliki fungsi mengendalikan tampilan karakter pada LCD. *register* dan memori merupakan komponen pelengkap yang terdapat pada mikrokontroler LCD. Berikut ini penjelasan dari gambar 2.6 :

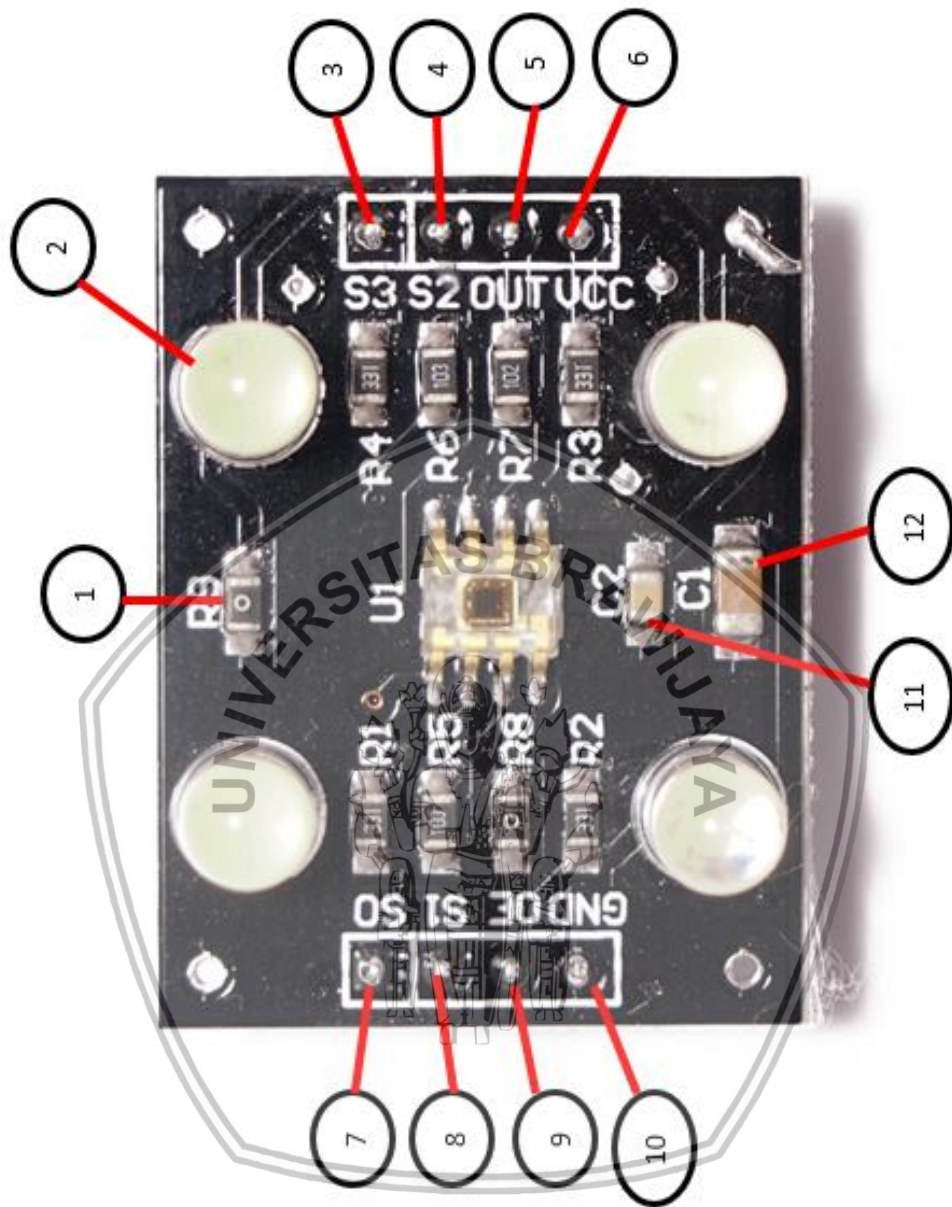
1. *VSS*, akhir dari tegangan yang masuk pada LCD.
2. *VDD*, Awal dari tegangan LCD.
3. *VO*, berfungsi sebagai pengatur kontras.
4. *RS*, berfungsi sebagai pembacaan *register*.
5. *R/W*, berfungsi sebagai pemacaan *read* dan penulisan *write* pada LCD.
6. *E*, digunakan sebagai pemanggil sinyal *Read* atau *Write*.
7. *DB*, berfungsi sebagai sinyal pengirim berupa bus data.
8. *Led +*, berfungsi sebagai tegangan positif pada LED.
9. *Led -*, berfungsi sebagai tegangan negatif pada LED.
10. *R*, berfungsi sebagai hambatan pada tegangan.
11. *Crystal*, berfungsi sebagai *Clock* pada IC16f628.
12. *RX* dan *TX*, berfungsi sebagai jalur komunikasi penerima RX dan pengirim data TX pada LCD.

### 2.2.5 Sensor TCS3200

Sistem berjalan dengan 2 sensor *photodiode* pada IC-nya. Sensor warna bekerja untuk pembacaan perbedaan warna pada minyak goreng. IC TCS3200 dapat membaca sistem warna dengan mengkonversikan nilai ADC dari output modul. Filter-filter tersebut didistribusikan pada masing-masing array. (Sutisna, 2015). Berikut ini merupakan contoh dari sensor TCS3200 yang ditunjukkan dengan Gambar 2.7.

Berikut ini merupakan fitur sensor warna TCS3200.

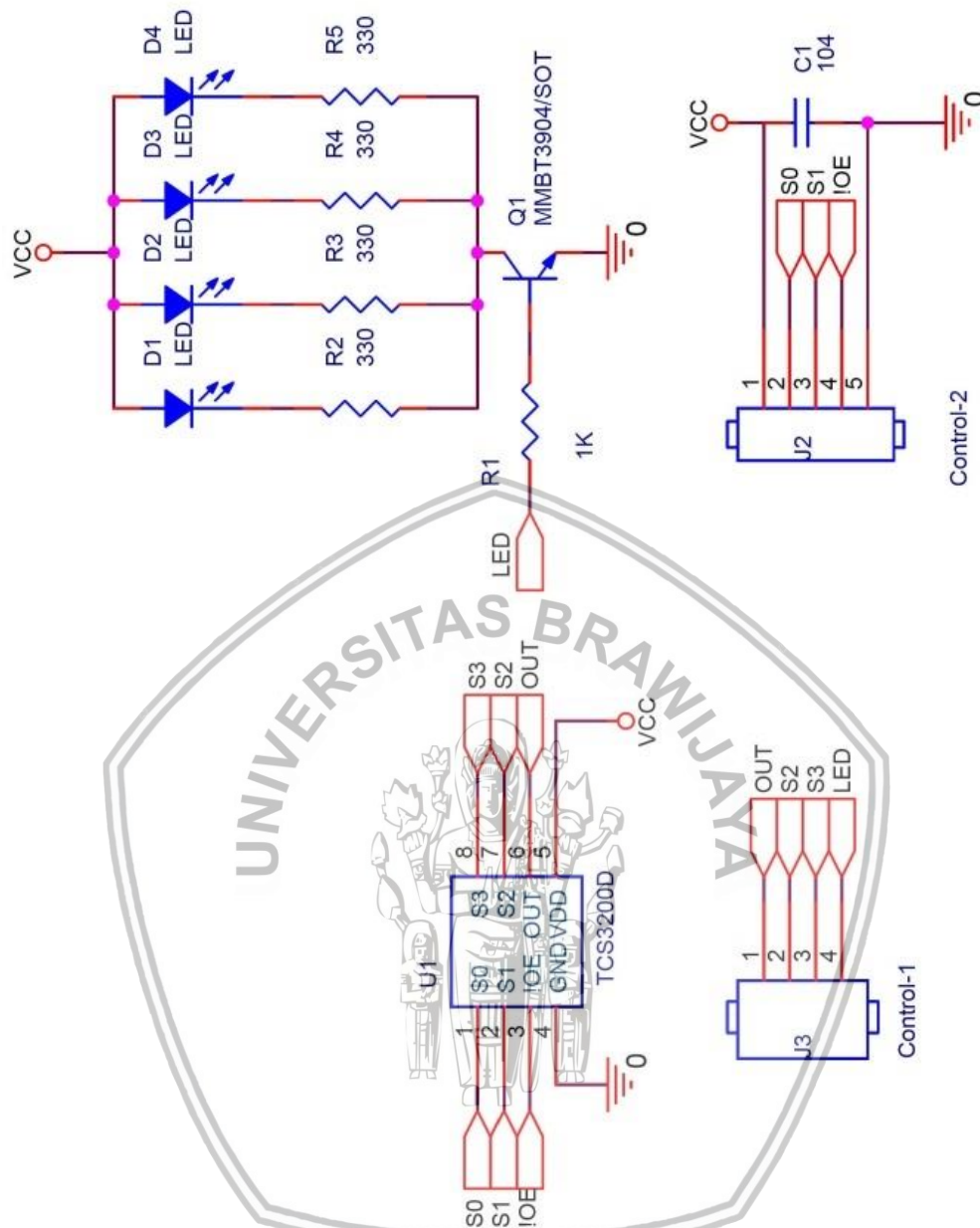
1. Resistor, berfungsi sebagai penghambat pada tegangan.
2. *Led*, berfungsi untuk pemancar cahaya menuju ke objek.
3. Pin *S3 inp* jenis *Photodiode* untuk membaca nilai cahaya yang masuk.
4. Pin *S2 input* jenis *Photodiode* untuk membaca nilai cahaya yang masuk.
5. *Out*, berfungsi sebagai *output* modul menuju ke mikrokontroler.
6. *VCC*, berfungsi sebagai sumber tegangan antara 2,7 sampai 5,5 Volt.
7. Pin *sinkronisasi S0* berfungsi sebagai skala input pilihan output frekuensi.
8. Pin *sinkronisasi S1* berfungsi sebagai skala input pilihan output frekuensi.
9. *E0*, enable sensor.
10. *Gnd*, berfungsi sebagai akhir dari jalannya tegangan yang masuk.



**Gambar 2.7 Sensor warna TCS3200**

Sumber: [elektronika-dasar.web.id](http://elektronika-dasar.web.id)





**Gambar 2.8 Skematik Sensor warna TCS3200**

Sumber : [www.dfrobot.com](http://www.dfrobot.com)

Gambar 2.8 menunjukkan skematik dalam dari sensor warna TCS 3200. Sensor tersebut berjalan dengan memancarkan cahaya LED yang pada LED tersebut terdapat rangkaian NPN untuk mengalirkan arus dari *Colector* menuju *Emiter* dengan mengaktifkan *bias*. Cahaya dari LED tersebut akan menabrak objek yang ada didepannya, kemudian cahaya akan memantul masuk menuju IC TCS3200 yang terletak ditengah modul. Alur system yang bekerja didalam IC tersebut dengan memanfaatkan 2 *photodiode* untuk membaca nilai cahaya masuk yang dipancarkan LED menuju objek. Kemudian hasil konfersi dari 2 *photodiode* tersebut menghasilkan 2 gelombang analog. Dari 2 gelombang analog tersebut kemudian di *convert* sebesar setengah gelombang. Sehingga *output* akhir dari IC bernilai 1,5 gelombang analog yang menjadi hasil keluaran berupa nilai merah,

hijau dan biru berupa nilai ADC yang kemudian dilakukan kalibrasi pada Arduino uno untuk mendapatkan hasil yang sesuai dengan nilai acuan warna pada *color tool photo-PAINT*. Tabel 2.2 menunjukkan pemilihan mode pengelompokan fotodiode pembacaan warna.

**Tabel 2.2 Pemilihan *mode photodiode* pada TCS3200**

S2	S3	Fotodiode
0	0	Merah
0	1	Biru
1	0	<i>Clear (no filter)</i>
1	1	Hijau

Saklar akan menentukan warna dengan sendirinya jika salah satu kelompok photodiode telah membaca input berupa intensitas cahaya terhadap objek yang terdeteksi oleh sensor. Kemudian Arduino sebagai mikrokontroler akan menginisialisasi nilai sensor TCS3200 yang dibaca oleh sensor. Nilai dari sensor selanjutnya dirubah menjadi frekuensi dengan mengubah arus yang menuju frekuensi. kemudian pada bagian tersebut terdapat osilator yang diaktifkan oleh saklar S0 serta S1 menjadi mode tegangan maksimal. Kemudian untuk membangkitkan tegangan minimal digunakan *output enable* yang akan membangkitkan osilator.

Berikut ini merupakan cara mengkalibrasi sensor warna TCS3200 yang digunakan pada penelitian ini dengan mendeteksi warna dari minyak goreng.

1. Masukkan gelas uji dengan memberikan kertas putih pada gelas uji tersebut.
2. Mengamati nilai random dari warna putih yang dideteksi.
3. Melakukan *mapping* pada kalibrasi sensor warna TCS3200 untuk mendapatkan nilai warna putih R 255, G 255, B 255.

```
void setup() {
  Serial.begin(9600);
  TSC_Init(); //inisialisasi sensor warna
  Timer1.initialize(); // inisialisasi library timer
  Timer1.attachInterrupt(TSC_Callback); //inisialisasi interrupt timer
  attachInterrupt(0, TSC_Count, RISING); //inisialisasi variabel interrupt
  delay(1000);
  pinMode(7, INPUT); //inisialisasi push button
  lcd.begin(16, 2); // set up the LCD's number of columns and rows:
  lcd.print("Power On"); // Print a message to the LCD.

  //kalibrasi sensor warna
  g_SF[0] = 255.0 / 440; //R Scale factor
  g_SF[1] = 255.0 / 450; //G Scale factor
  g_SF[2] = 255.0 / 510; //B Scale factor
}
```

**Gambar 2.9 Bagian *source code* yang diubah**

Sumber : M Nuzulul Marofi, (2017)

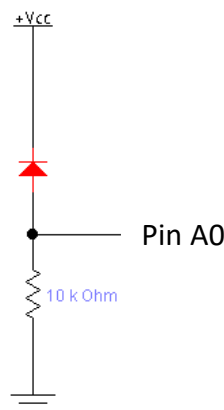
4. Meng-upload program yang sudah dikalibrasi.
5. Dilakukan secara terus menerus sampai nilai pembacaan sensor hampir sama dengan *color tool photo-Phaint*.

### 2.2.6 Sensor *Photodiode*

*Photodiode* merupakan komponen yang terbuat dari bahan berupa semikonduktor yang menyerupai LED dan dapat mengubah intensitas cahaya yang masuk pada *photodiode* menjadi arus listrik. Ada 2 jenis *photodiode* yang dijual di toko-toko elektronik yaitu *photodiode* dengan warna bening dan warna hitam. Pada penelitian ini sensor photodiode yang digunakan sebagai pendeteksi kekeruhan adalah *photodiode* dengan warna bening. Karena proses pembacaan *photodiode* dengan LED bening sesuai dengan hasil output pada jurnal sebelumnya. Pada skematik *photodiode*, arah panah pada LED photodiode masuk yang dapat diartikan LED tersebut menerima intensitas cahaya yang kemudian dikonfersikan menjadi sebuah tegangan (Winarno, 2015).

Arus dari tegangan akan melewati resistor. Tegangan yang timbul dari penerimaan cahaya yang masuk pada photodiode inilah yang kemudian masuk pada pin A0 pada Arduino dengan nilai yang sebanding dengan intensitas cahaya. Resistor yang digunakan pada alat memiliki ukuran 10k ohm dan 220 ohm. Penggunaan Resistor dengan ukuran 10k ohm berfungsi sebagai pembagi tegangan fotodioda yang digunakan. Kemudian untuk dapat menyalakan LED digunakan resistor dengan kapasitas 220 ohm. Skematik rangkaian *photodiode* dapat dilihat pada Gambar 2.10 dan penggambaran fisik LED dapat dilihat pada Gambar 2.11.

Fungsi dari sensor yaitu dengan menerima intensitas cahaya LED. cahaya LED ditembakkan ke arah minyak goreng sehingga, cahaya dapat menembus dan dapat terbaca oleh sensor fotodioda. sehingga sensor akan bekerja sebagai sumber tegangan yang bermuatan arus. Sehingga arus yang dikeluarkan oleh sensor akan mengalir didalam rangkaian. Data keluaran dari sensor fotodioda berupa nilai tegangan yang berfungsi untuk mendeteksi minyak goreng berdasarkan tingkat kekeruhannya.



**Gambar 2.10 Skematik *Photodiode***

Sumber : Winarno, 2011 disitasi dalam Monica, (2015)



**Gambar 2.11 Gambar fisik LED *Photodiode***

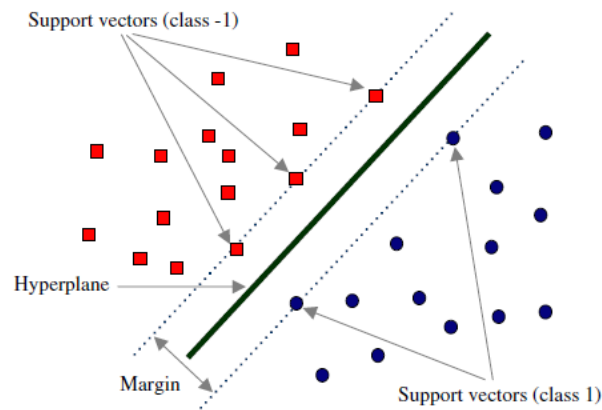
Sumber: lapantech.com

### 2.2.7 Klasifikasi

Klasifikasi adalah proses dimana terjadi pengelompokan suatu data kedalam sebuah kelas yang sudah ditentukan dan berdasar kepada ciri, sifat, dan pola yang terdapat pada tiap-tiap kelas tersebut. Klasifikasi bekerja dengan cara memperkenalkan kelompok atau sekumpulan data sebagai pedoman kriteria dalam sebuah kelas, untuk kemudian pedoman tersebut digunakan untuk memilah-milah setiap data yang cocok dengan kriteria salah satu dari kelas yang sudah ditentukan. Klasifikasi dibagi menjadi 2 dalam proses fasenya (Pramudiono, 2003), yaitu fase *learning* dan *testing*. Ada berbagai macam metode yang digunakan untuk melakukan proses klasifikasi, yaitu antara lain *K-Nearest Neighbor*, *Support Vector Machine (SVM)*, *Naïve Bayes*, dll.

### 2.2.8 Support Vector Machine

SVM sendiri merupakan metode *learning machine* yang bertujuan mendapatkan batas atau jarak terbaik yang dapat memisahkan antara 2 kelas, juga dapat disebut dengan istilah *hyperplane* (Satriyo, 2003). SVM berada dalam satu kelas dengan *Artificial Neural Network (ANN)* dalam hal fungsi dan kondisi permasalahan yang bisa diselesaikan. Keduanya masuk dalam kelas supervised learning. Cara kerja metode SVM dengan memisahkan anggota dari 2 buah class 1 dan -1. Warna merah yang terdapat pada gambar 2.12 merupakan *pattern* dengan kelas -1, kemudian warna biru digunakan untuk menggambarkan *pattern* pada kelas 1. Untuk memecahkan masalah tersebut dapat diterjemahkan dengan cara menentukan garis pembatas atau *hyperplane* yang dijadikan sebagai pemisah antar dua kelompok kelas.



**Gambar 2.12 Hyperplane SVM**

Sumber : amitranga.wordpress.com

Garis pemisah terbaik antara 2 kelas yang berbeda atau disebut juga *hyperplane* dapat ditemukan nilainya dengan mencari nilai yang paling terdekat antara kelas 1 dengan *hyperplane* dan juga nilai yang paling terdekat antara kelas -1 dengan *hyperplane*. Jarak atau *margin* terdekat antara *pattern* dengan *hyperplane* inilah yang menjadi nilai penentu pemisah kelas terbaik. Pemisah yang berisi nilai *pattern* paling dekat inilah yang disebut dengan istilah *support vector*. Garis yang melintang pada Gambar 2.12 menunjukkan *hyperplane* yang terbaik dan tepat pada tengah-tengah dari 2 kelas yang berbeda, sedangkan warna biru dan merah yang berada didalam Gambar 2.12 merupakan *support vector*. Proses dalam pencarian nilai *hyperplane* yang terbaik inilah yang dilakukan pada penelitian ini dengan menerapkan metode SVM.

### 2.2.8.1 Support Vector Machine (SVM) Linier

SVM *Linier* adalah suatu langkah untuk menggolongkan dua buah kelas dengan memisahnya menggunakan data secara linier. Dengan menggunakan fungsi berupa  $f(w,b)=x_i \cdot w + b$  untuk menemukan kelas pemisah dengan hasil 1 dan -1.

$$w \cdot x_i + b \geq +1 \text{ untuk } y_i = 1 \quad (2.1)$$

$$w \cdot x_i + b \leq -1 \text{ untuk } y_i = -1 \quad (2.2)$$

Penjelasan :

$x_i$  = inialisasi data ke-i.

$w$  = nilai support vector yang sebanding dengan hyperplane.

$b$  = nilai hasil bias.

$y_i$  = data ke- $i$  yang ekuivalen dengan persamaan 2.3.

$$y_i (x_i \cdot w + b) - 1 \geq 0 \text{ untuk } i = 1, \dots, n. \quad (2.3)$$

Penjelasan :

$n$  = nilai seluruh sampai data ke  $n$ .



### 2.2.8.2 Support Vector Machine (SVM) Non-Linier

SVM *non-linier* digunakan untuk memecahkan permasalahan *non-linier* dengan menambahkan fungsi *Kernel*. Pada SVM *non-Linier*, nilai awal  $x$  yang dimasukkan pada fungsi  $\Phi(x)$  menuju sebuah ruangan *vector* yang memiliki dimensi lebih tinggi (Nugroho, et al., 2003). Dibawah ini merupakan berbagai macam fungsi *kernel* yang terdapat pada SVM :

1. Fungsi Linier, memiliki persamaan  $K(x, y) = x \cdot y$  (2.4)

2. Fungsi Polinomial of degree, memiliki persamaan  $K(x, y) = (x \cdot y)^d$  (2.5)

3. Fungsi Polinomial of degree 2, dengan persamaan  $K(x, y) = (x \cdot y + c)^d$  (2.6)

4. Fungsi gaussian RBF; memiliki persamaan  $K(x, y) = \exp\left(\frac{-\|x-y\|^2}{2\sigma^2}\right)$  (2.7)

Penjelasan:

$K(x, y)$  = fungsi *kernel*.

$x_i$  = nilai data ke  $i$ .

$x_j$  = nilai data ke  $j$ .

$c$  = nilai konstanta  $C$ .

$\sigma$  = *sigma* terkecil dari *kernel* RBF.

### 2.2.8.3 Sequential Training

SVM mempunyai berbagai proses *training* berupa *Sequential Minimal*, *Sequential Training*, dan *Optimization Quadratic Programming*. Pada *Quadratic Programming* membutuhkan waktu pemrosesan cukup lama dan memiliki algoritma yang kompleks. Sedangkan perkembangan dari *Quadratic Programming* disebut juga dengan *Sequential Minimal Optimization*, pada proses ini juga memiliki algoritma yang kompleks namun hanya berfokus pada permasalahan optimasi yang kecil. Pada proses *Sequential Training*, algoritma yang digunakan lebih sederhana, serta tidak memakan waktu pemrosesan yang lama. Dibawah ini merupakan algoritma *Sequential Training* (Wu, 1999).

1. Inisialisasi pada  $\sigma_1$  lalu parameter selanjutnya, seperti  $\lambda, \gamma, C, \varepsilon$ .

Keterangan :

$\sigma_1$  = *sigma*.

$\alpha_1$  = *alfa*, berfungsi untuk inisialisasi awal *support vector*.

$\gamma$  = *gamma* untuk mencari nilai kecepatan.

$C$  = untuk memberi batas nilai *alfa*.

$\varepsilon$  = *epsilon* berfungsi menemukan nilai *error*.

2. Menemukan nilai matrik *Hessian* dengan menggunakan persamaan.

$$D_{ij} = y_i y_j (K(x_i, x_j) + \lambda^2) \quad (2.8)$$



Dengan  $i, j : 1, \dots, n$ .

Keterangan :

$x_i$  = data nilai ke- $i$ .

$x_j$  = data nilai ke- $j$ .

$y_i$  = data nilai dari kelas ke- $i$ .

$y_j$  = data nilai dari kelas ke- $j$ .

$n$  = jumlah seluruh sampai data ke  $n$ .

$K(x_i, x_j)$  = *kernel* nilai yang dipakai.

3. Dimulai dari nilai data  $i$  sampai dengan data  $j$ , dijelaskan dalam persamaan dibawah ini:

$$a. E_i = \sum_{j=1}^n \alpha_j D_{ij} \quad (2.9)$$

Keterangan :

$\alpha_j$  = nilai *alfa* ke- $j$ .

$D_{ij}$  = Matrik *Hessian*.

$E_{ij}$  = rata-rata *Error*.

$$b. \delta \alpha_i = \min(\max[\gamma(1 - E_i), \alpha_i], C - \alpha_i) \quad (2.10)$$

Keterangan :

$\alpha_i$  = nilai *alfa* ke- $i$ .

$\gamma$  = *gamma* untuk mencari nilai kecepatan.

$E_{ij}$  = rata-rata *Error*.

$C$  = untuk memberi batas nilai *alfa*.

$$c. \alpha_i = \alpha_i + \delta \alpha_i \quad (2.11)$$

Keterangan :

$\alpha_i$  = *alfa* nilai ke- $i$ .

$\delta \alpha_i$  = delta *alfa* nilai ke- $i$ .

4. Proses ke 3 diulang terus sampai mendapatkan nilai iterasi maksimal ( $\delta \alpha_i \leq \varepsilon$ ).

5. Selanjutnya mendapatkan nilai *support vector*,  $SV = (\alpha_i > \text{thresholdSV})$

misalkan *threshold SV* = 0.

#### 2.2.8.4 Testing SVM

1. Langkah awal perhitungan nilai  $f(x)$ , dengan menggunakan rumus dibawah ini:

$$f(x) = \sum_{i=1}^m \alpha_i y_i K(x_i, x) + b \quad (2.12)$$

Keterangan :

- $\alpha_i$  = *alfa* nilai ke-*i*.  
 $y_i$  = data nilai dari kelas ke-*i*.  
 $m$  = data jumlah dari SV.  
 $K(x_i, x_j)$  = *kernel* nilai yang dipakai.  
 $b$  = nilai *bias*.

Dibawah ini merupakan rumus untuk mencari nilai *bias* :

$$b = -\frac{1}{2} [\sum_{i=1}^m \alpha_i y_i K(x_i, x^+) + \sum_{i=1}^m \alpha_i y_i K(x_i, x^-)] \quad (2.13)$$

Penjelasan :

- $\alpha_i$  = *alfa* nilai ke-*i*.  
 $y_i$  = data nilai dari kelas ke-*i*.  
 $m$  = data jumlah dari SV.  
 $K(x_i, x^-), K(x_i, x^+)$  = *kernel* nilai yang dipakai.

2. Kemudian mencari nilai  $K(x_i, x_{test})$

$$\sum_{i=1}^m y_i y_j K(x_i, x_j) \quad (2.14)$$

Keterangan :

- $y_i$  = data kelas ke-*i*.  
 $y_j$  = data kelas ke-*j*.  
 $m$  = jumlah data dari SV.  
 $K(x_i, x)$  = Penggunaan *kernel*.

3. Langkah terakhir mencari nilai  $f(x_{test})$

$$f(x_{test}) = \sum_{i=1}^m \alpha_i y_i K(x_i, x) + b \quad (2.15)$$

Keterangan :

- $y_i$  = kelas dari data ke-*i*.  
 $y_j$  = kelas dari data ke-*j*.  
 $m$  = jumlah data dari SV.  
 $K(x_i, x)$  = penggunaan *kernel*.

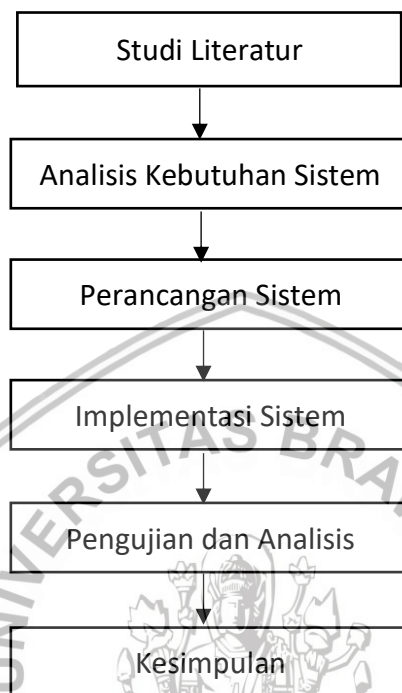
## 2.2.9 Evaluasi

Mengukur kinerja suatu system klasifikasi yang mana menggambarkan hasil dari proses klasifikasi menunjukkan akurasi yang baik atau tidak, sehingga didapatkan nilai hasil akurasi. nilai akurasi tersebut menunjukkan tingkat akurasi dari sistem dalam mengelompokan data sesuai aturan yang sudah dibuat dengan menggunakan persamaan 2.16.

$$Akurasi = \frac{\text{jumlah data benar}}{\text{seluruh data}} * 100\% \quad (2.16)$$

## BAB 3 METODOLOGI

Pada bab ini menunjukkan tentang beberapa tahapan metodologi penelitian yang ditunjukkan pada Gambar 3.1 berupa diagram blok :



Gambar 3.1 Diagram alur penelitian

### 3.1 Studi Literatur

Pada studi literature dilakukan pengupulan informasi dan sumber-sumber yang digunakan sebagai acuan dalam penelitian yang akan dilaksanakan. Pada penelitian ini yaitu dengan pembuatan system Implementasi Metode Klasifikasi *Support Vector Machine* (SVM) Terhadap Pemakaian Minyak Goreng. Referensi tersebut diperoleh dari beberapa sumber seperti jurnal, buku, dan bimbingan yang diberikan dosen pembimbing. literature yang dipelajari sebagai berikut :

1. Kajian pustaka.
2. Kualitas minyak goreng.
3. Arduino uno.
4. Sensor dan tranduser.
5. Klasifikasi.
6. *Machine learning*.
7. Algoritma *Support Vector Machine* (SVM).

### 3.2 Analisis Kebutuhan Sistem

Pada bagian kebutuhan sistem bertujuan mengetahui apa saja yang dibutuhkan untuk penelitian dan tidak menyimpang jauh dari permasalahan yang akan diselesaikan. Tahapan tersebut meliputi analisis dari kriteria system dari berbagai kebutuhan yang digunakan oleh peneliti. Berikut merupakan alur dan komponen yang digunakan dalam sistem implementasi metode klasifikasi *Support Vector Machine* (SVM) terhadap pemakaian minyak goreng.

#### 3.2.1 Kebutuhan Perangkat Keras

Berikut ini merupakan kebutuhan perangkat keras *system* pada penelitian, meliputi :

1. Model perangkat : HP Pavilion Sleekbook 14 PC, Intel(R) Core(TM) i3-2367M CPU @ 1.40GHz (4 CPUs), *Memory* : 4GB RAM.
2. Mikrokontroler Arduino Uno.
3. LCD (*Liquid Crystal Display*).
4. Sensor warna TCS3200.
5. Sensor *photodiode*.
6. 1 unit *mini Breadboard*.

#### 3.2.2 Kebutuhan Perangkat Lunak

Berikut ini merupakan kebutuhan perangkat keras *system* pada penelitian, meliputi :

1. *Windows 10*, untuk menjalankan perangkat lunak.
2. *Arduino IDE*, untuk membuat program yang dibutuhkan pada pembuatan sistem.
3. *Python*, untuk melakukan pemrograman bahasa *Python*.

### 3.3 Perancangan Sistem

Dalam merancang system untuk implementasi metode klasifikasi *Support Vector Machine* (SVM) terhadap pemakaian minyak goreng, memiliki beberapa tahapan. Tahapan yang pertama adalah melakukan pengambilan data yang dilakukan pada *prototype* kemudian melakukan normalisasi pada data latih dan data uji. Selanjutnya menentukan fungsi *kernel* yang dipakai pada algoritma SVM, lalu melakukan proses *testing SVM* dan melakukan tahap klasifikasi terhadap penggunaan minyak goreng.

### 3.4 Implementasi Sistem

Dalam tahap implementasi *system*, peneliti membangun sebuah system dengan menerapkan beberapa hal yang didapat dari proses literature. Setelah

mendapatkan pandangan dari studi literature, proses tersebut diimplementasikan dengan berbagai tahap seperti yang dijelaskan dibawah ini :

1. Peng-implementasian pada alat pengujian minyak goreng. Dilakukan pembuatan sebuah *prototype* sebagai wadah yang digunakan untuk menaruh gelas uji yang sebelumnya sudah dirancang oleh peneliti dengan tujuan mempermudah penggunaan alat dan sensor dapat berjalan dengan baik.
2. Implementasi komponen penunjang penelitian berupa sensor-sensor yang dibutuhkan dengan peletakan pada tempat yang sudah direncanakan. Hal ini bertujuan agar komponen sensor-sensor yang digunakan tersebut dapat bekerja dengan optimal pada saat proses pengujian.
3. Implementasi pada perangkat lunak, pada tahap ini menjelaskan tentang pengaplikasian system pada penelitian dengan menggunakan Bahasa pemrograman Arduino IDE untuk mempermudah kalibrasi dan pembacaan sensor dan juga bahasa pemrograman python sebagai pengolah nilai matrik untuk menentukan kelas dari data yang telah dilatih dan data uji.
4. Implementasi metode yang digunakan dalam penelitian, yaitu metode *Support Vector Machine*. Untuk pengklasifikasian kelas dalam penggunaan minyak goreng digunakan metode SVM.

### 3.5 Pengujian dan Analisis

Pada tahap pengujian system berupa *learning machine* pada data latih yang sudah ditentukan kriteria didalamnya untuk kemudian dilakukan pengujian dengan mencocokkan data uji yang masuk dengan data latih yang sudah ada pada lembar CSV.

1. Proses pengujian sensor warna TCS3200.
2. Proses pengujian sensor fotodiode.
3. Proses pengujian rasio perbandingan data.
4. Proses pengujian kinerja *kernel*.
5. Proses pengujian *Sequential Training* pada SVM sebagai parameter uji :
  - a) Proses pengujian dengan menggunakan parameter nilai  $\lambda$ .
  - b) Proses pengujian dengan menggunakan parameter nilai  $C$  ( *Complexity* ).
  - c) Proses pengujian dengan menggunakan parameter nilai *itermax*.
6. Proses pengujian *itermax*.
7. Pengujian keakuratan sistem klasifikasi penggunaan minyak goreng menggunakan metode *Support Vector Machine* dengan membandingkan data latih pada CSV dengan data yang diujikan berupa nilai *array* dari sensor.

Analisis yang dilakukan adalah dengan mengukur tingkat hasil akurasi system yang diproses dengan langsung membaca data latih dan yang satunya

diproses dengan input berupa nilai array dari sensor. Menghitung tingkat akurasi dapat dihitung dengan persamaan 2.16.

### 3.6 Kesimpulan dan Saran

Pada bab ini dilakukan penarikan kesimpulan dari keseluruhan proses yang telah dilakukan mulai dari tahap perancangan, implementasi dan pengujian. Kesimpulan digunakan untuk mengetahui keberhasilan penelitian sesuai dengan rumusan masalah yang telah di sebutkan pada bab pendahuluan penelitian. Kemudian untuk mempermudah penelitian yang akan dilakukan dikemudian hari diberikan saran dari hasil penelitian yang telah dilakukan. Sehingga mempermudah dalam pengembangan dan penyempurnaan pada penelitian dikemudian hari.





## BAB 4 REKAYASA KEBUTUHAN

Pada bab ini penelitian yang berjudul “Implementasi Metode Klasifikasi *Support Vector Machine* (SVM) Terhadap Pemakaian Minyak Goreng”, berisi tentang deskripsi tujuan dari sistem, manfaat sistem yang dibuat, karakteristik pengguna yang nantinya akan menggunakan sistem, lingkungan dimana sistem dapat beroperasi, batasan dari sistem, serta asumsi dan ketergantungan dari sistem tersebut.

### 4.1 Deskripsi Umum

Sistem ini dapat mengklasifikasi penggunaan minyak goreng dengan menggunakan minyak goreng kemasan dan curah dengan objek yang digoreng berupa ikan dan non ikan. Klasifikasi sendiri berdasarkan pada nilai yang dibaca oleh sensor warna TCS3200 sebagai inputan dengan fitur RGB dan sensor fotodiode dengan fitur P.

Sistem ini dirancang menggunakan metode *Support Vector Machine*, sebagai metode untuk pengklasifikasian minyak goreng masuk kedalam kelas dengan objek penggorengan ikan atau non ikan. Terdapat 4 inputan untuk mengambil data yaitu sensor TCS3200 sebagai alat pengukur warna pada minyak goreng. Kemudian sensor fotodiode sebagai alat pengukur kekeruhan dari minyak goreng tersebut. Setelah itu sistem membedakan 2 kelas objek penggorengan untuk dapat diketahui bahwa minyak goreng tersebut digunakan untuk menggoreng ikan atau non-ikan. Dari tiap kelas akan diklasifikasikan lagi dengan perhitungan SVM untuk menentukan minyak goreng tersebut masih layak digunakan atau tidak dengan melihat nilai  $f_x$  dari tiap perbandingan data uji dengan data latih yang sebelumnya sudah dimasukkan kedalam perhitungan metode. Dengan parameter penggunaan minyak maksimal 2 kali penggorengan sebagai acuan kelayakan minyak goreng dan juga analisis warna dan kekeruhan minyak secara manual untuk memastikan minyak masih layak dipakai ataupun tidak. Output yang digunakan untuk mengetahui hasil *learning machine sistem* adalah LCD 16x2 dengan tampilan kelayakan minyak goreng tersebut beserta objek penggorengan sebelumnya.

### 4.2 Rekayasa Kebutuhan

Pada bab ini akan menguraikan semua kebutuhan yang memiliki tujuan agar implementasi metode klasifikasi *Support Vector Machine* (SVM) terhadap pemakaian minyak goreng dapat bekerja sesuai dengan tujuan awal yang terdiri dari kebutuhan fungsional, kebutuhan komunikasi dan kebutuhan lain yang dibutuhkan oleh sistem dan akan dijelaskan pada sub bab dibawah ini.

#### 4.2.1 Kebutuhan Antarmuka

Antarmuka pada sistem ini akan ditampilkan pada LCD 16x2. Antarmuka ini digunakan oleh user untuk menampilkan *input*, kemudian proses perhitungan

metode *Support Vector Machine* dan *output* berupa hasil klasifikasi dari minyak goreng tersebut yang kemudian ditampilkan pada LCD 16x2.

#### 4.2.2 Kebutuhan Perangkat Keras

Berikut ini merupakan kebutuhan perangkat keras yang dibutuhkan oleh sistem :

1. Laptop

Dalam hal ini, laptop digunakan untuk membuat program dalam sistem dan sebagai sumber daya dari sistem. Laptop yang digunakan adalah HP Pavilion Sleekbook 14 dengan prosesor Intel Core i3 CPU 1,40 GHz, serta menggunakan sistem operasi Windows 10 Pro 64-bit.

2. Arduino Uno

Mikrokontroler Arduino Uno berfungsi sebagai pengolah inputan atau dapat disebut sebagai otak dalam sistem.

3. Sensor warna TCS3200

Sensor warna TCS3200 berfungsi sebagai pendeteksi warna dari minyak goreng yang diuji.

4. Sensor fotodiode

Sensor fotodiode berfungsi sebagai pendeteksi tingkat kekeruhan dari minyak goreng yang diuji.

5. LCD 16 x 2

LCD 16x2 digunakan untuk menampilkan hasil perhitungan sistem.

6. Kabel Jumper

Kabel jumper digunakan untuk menghubungkan antara perangkat satu dengan perangkat lainnya.

7. *Push Button*

*Push Button* digunakan sebagai *trigger* untuk pengambilan data, kemudian data akan diproses dengan menggunakan metode *Bayes*.

8. *I2C LCD*

*I2C LCD* digunakan sebagai pengatur intensitas daya yang masuk pada LCD sebagai pengontrol kecerahan dan juga meminimalisir pin LCD yang masuk pada Arduino dengan menggunakan *modul I2C*.

Pengguna yang menggunakan sistem ini adalah masyarakat luas, khususnya masyarakat (Pedagang Kaki Lima) yang sering melakukan kegiatan penggorengan menggunakan minyak goreng dan minyak goreng tersebut digunakan secara berulang. Biasanya masyarakat yang menggunakan minyak goreng berulang adalah masyarakat yang bisa dikatakan perekonomiannya termasuk golongan menengah dan menengah ke bawah.

### 4.2.3 Kebutuhan Perangkat Lunak

Perangkat lunak ini digunakan untuk memberi dukungan pada perangkat keras yang ada. Walaupun perangkat keras sudah lengkap dan memenuhi syarat dalam pembuatan sistem, tidak akan berjalan tanpa adanya perangkat lunak ini. Berikut ini merupakan perangkat lunak yang dibutuhkan oleh sistem :

1. Windows 10 Pro 64-bit

Sistem operasi Windows 10 Pro 64-bit digunakan sebagai sistem operasi pada laptop yang digunakan untuk menyelesaikan penelitian ini.

2. *Python*

Pemrograman *python* digunakan sebagai penghitung data latih berupa nilai matrik *Hassien* dengan total data latih sebanyak 96 dan data uji sebanyak 15. Kemudian hasil *learning* dari metode diteruskan menuju Arduino untuk dilakukan proses selanjutnya yaitu menampilkan hasil perbandingan data uji dengan *learning* data latih pada *python*.

3. Arduino IDE

Arduino IDE digunakan untuk mengoperasikan Arduino, membuat program pada sistem, selain itu terdapat fitur *library* pada Arduino IDE berguna dalam memudahkan pembuatan program pada sistem.

4. Library Liquid Crystal I2C.h

Library liquid crystal I2C.h digunakan untuk memanggil modul LCD yang dikendalikan secara serial sinkron dengan protokol *Inter Integrated Circuit*. Dikendalikan secara paralel dengan hanya menggunakan 2 pin digital.

5. Library Wire.h

Wire.h digunakan untuk memanggil modul LCD yang dikendalikan secara serial sinkron dengan protokol *Two Wire Interface*. Merujuk pada protokol komunikasi serial sinkron dengan 2 kabel yakni SDA (Serial Data) dan SCL (Serial Clock). SDA merupakan data serial, sedangkan SCL adalah jalur clock sinkronasinya.

6. CSV

CSV digunakan untuk menyimpan file data latih berupa .excel maupun .csv yang memiliki batas nilai berupa tanda koma(,).

7. *Simplejson*

*Simplejson* digunakan pada Arduino IDE dan juga Python sebagai format pertukaran data yang ringan, dan mudah dibaca oleh manusia serta diterjemahkan oleh computer.

#### 4.2.4 Kebutuhan Komunikasi

Arduino UNO mempunyai sejumlah fasilitas untuk komunikasi dengan sebuah komputer, Arduino atau mikrokontroler lainnya. Arduino Uno menyediakan serial komunikasi UART TTL (5V), yang tersedia pada pin digital 0 (RX) dan 1 (TX). Sebuah Atmega 16U2 pada channel board serial komunikasinya melalui USB dan muncul sebagai sebuah port virtual menuju software pada komputer. Firmware 16U2 menggunakan driver USB COM standar, dan tidak ada driver eksternal yang dibutuhkan. LED RX dan TX pada board akan menyala ketika data sedang ditransmit melalui chip USB-to-serial dan koneksi USB pada komputer (tapi tidak untuk komunikasi serial pada pin 0 dan 1). Arduino Uno juga mensupport komunikasi I2C (TWI) dan SPI.

#### 4.2.5 Kebutuhan Fungsional

Pada kebutuhan fungsional menjelaskan tentang bagaimana sistem berjalan sesuai yang diharapkan dan sesuai dengan tujuan penelitian. Apabila kebutuhan fungsional ini tidak terpenuhi maka sistem tidak berjalan dengan baik atau dapat dikatakan sistem dapat mengalami kegagalan.

##### 4.2.5.1 Fungsi System Pendeteksi Warna

Fungsi ini mengharuskan pin sensor warna TCS3200 dapat tersambung ke mikrokontroller Arduino UNO. Dengan mengatur pin sensor S0, S1, S2, S3 dan juga Out ke pin 7 pada Arduino supaya sensor dapat membaca dan mengirim data dan diimplementasikan pada sistem dengan outputan ditampilkan pada LCD dengan data berupa nilai warna yang telah dikalibrasi sebelumnya terhadap minyak goreng.

##### 4.2.5.2 Fungsi System Pendeteksi Kekeruhan

Fungsi ini mengharuskan pin sensor fotodiode dapat tersambung ke mikrokontroller Arduino UNO. Dengan mengatur pin sensor Out ke pin A0 pada Arduino supaya sensor dapat membaca dan mengirim data dan diimplementasikan pada sistem dengan outputan ditampilkan pada LCD dengan data berupa nilai kekeruhan yang telah dikalibrasi sebelumnya terhadap minyak goreng.

##### 4.2.5.3 Fungsi Klasifikasi Menggunakan Metode SVM

Fungsi ini mengharuskan sistem mampu menentukan kondisi sesuai dengan input yang diterima oleh sensor dan diolah pada *python* yang hasilnya digunakan untuk menentukan kondisi minyak goreng yang diuji. Fungsi ini merupakan fungsi yang diprioritaskan karena merupakan tujuan utama dari sistem yaitu sistem mampu menentukan output yang akan dijalankan.

#### 4.2.5.4 Fungsi Tampilkan Hasil

Fungsi ini mengharuskan system mampu menampilkan berbagai macam kondisi yang diinginkan oleh pengguna untuk kemudian ditampilkan pada LCD. Fungsi ini merupakan fungsi yang diprioritaskan karena merupakan tujuan utama dari system yaitu system mampu menampilkan hasil kondisi pada LCD.

#### 4.2.6 Kebutuhan Non Fungsional

Berikut ini merupakan kebutuhan non fungsional dari sistem :

1. Alat dibuat dengan model yang semudah mungkin untuk dapat digunakan oleh user.
2. Penggunaan metode SVM untuk pemetaan dalam beberapa kondisi minyak goreng.
3. Sistem alat dibuat secara portable dengan menggunakan bahan bekas yang didesain sebaik dan semudah mungkin untuk dapat digunakan dimanapun.

#### 4.3 Batasan Desain Sistem

Implementasi metode klasifikasi *Support Vector Machine* (SVM) terhadap pemakaian minyak goreng dilakukan agar sesuai dengan hasil yang diharapkan, maka diperlukan batasan-batasan implementasi desain sistem. Batasan perancangan dan implementasi pada sistem dapat dilihat di bawah ini :

1. Sistem ini hanya dapat mengklasifikasikan frekuensi penggunaan minyak goreng dengan menggunakan bahan ikan dan tahu.
2. Sistem hanya menggunakan sensor warna TCS3200 dan sensor fotodioda masing-masing satu buah.
3. Sistem hanya menggunakan gelas pengujian yang sudah disiapkan oleh peneliti.
4. Sistem hanya menggunakan metode *Support Vector Machine* untuk menentukan hasil klasifikasi frekuensi penggunaan minyak goreng.

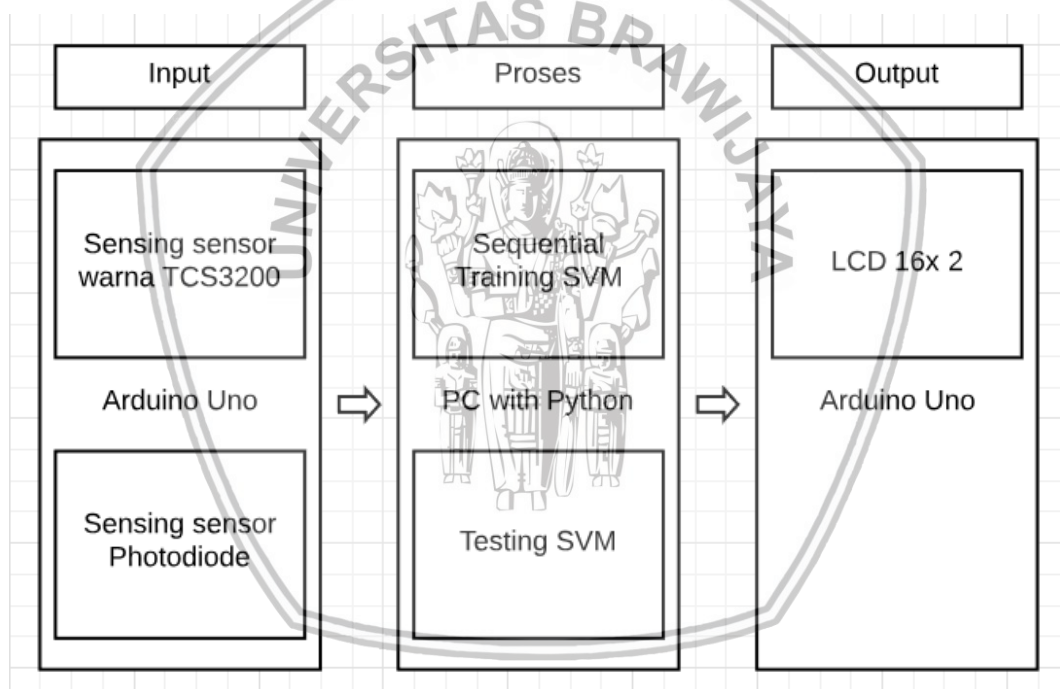


## BAB 5 PERANCANGAN DAN IMPLEMENTASI

Bab ini membahas mengenai perancangan dan implementasi dari sistem yang akan dibuat untuk penelitian yang akan dilakukan yaitu “Implementasi Metode Klasifikasi *Support Vector Machine* (SVM) Terhadap Pemakaian Minyak Goreng”. Pada perancangan terdapat perancangan perangkat keras (hardware) dan perancangan perangkat lunak (software). Sedangkan pada implementasi sistem terdapat implementasi perangkat keras (hardware) dan implementasi perangkat lunak (software).

### 5.1 Perancangan Sistem

Perancangan sistem terdiri dari 3 bagian, yaitu input berupa sensor TCS3200 dan fotodiode. Proses berupa perhitungan klasifikasi metode SVM. Output berupa hasil klasifikasi yang ditampilkan pada LCD 16 x 2 ditunjukkan pada Gambar 5.1.



Gambar 5.1 Diagram blok sistem

Gambar 5.1 terdiri dari beberapa blok diagram proses, yaitu :

1. *Input*

*Input* pada sistem ini berupa dua parameter yaitu sensor TCS3200 sebagai *input* warna dan sensor fotodiode sebagai *input* kekeruhan minyak goreng yang dilakukan pengkalibrasian nilai pembacaan sensor awal pada Arduino.

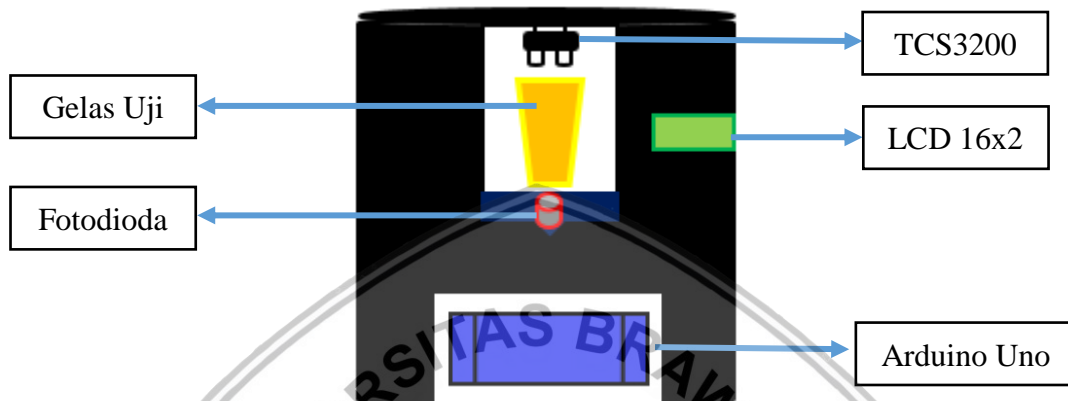
2. *Proses*

Pada bagian proses, Arduino bertugas sebagai pengambil input untuk kemudian dilakukan proses perhitungan algoritma *Support Vector Machine*

(SVM) pada laptop dengan menggunakan pemrograman *python* sebagai pengolah nilai dari data latih berupa nilai matrik *Hassian*.

### 3. Output

Keputusan klasifikasi penggunaan minyak goreng yang mempunyai kesamaan nilai dengan kelas yang sudah ditentukan sebelumnya dan kemudian hasilnya ditampilkan pada LCD 16x2.



Gambar 5.2 Desain Perancangan Sistem

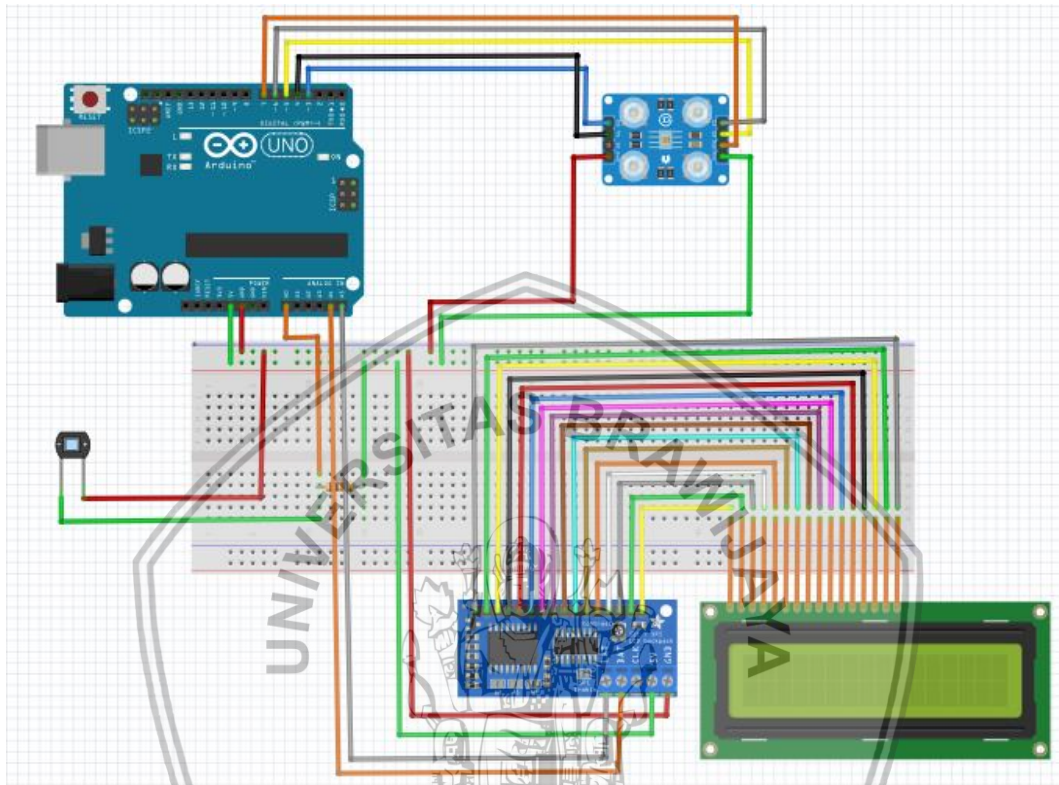
Pada bagian ini berisi mengenai perancangan dan implementasi pada penelitian “Implementasi Metode Klasifikasi *Support Vector Machine* (SVM) Terhadap Pemakaian Minyak Goreng” yang ditunjukkan pada Gambar 5.2. Pada perancangan sistem akan dijelaskan bagaimana sistem dibangun agar mampu mendeteksi penggunaan minyak goreng dengan objek penggorengan berupa ikan maupun non ikan dengan baik. Pada bagian perancangan terdapat dua perancangan sistem, yaitu perancangan perangkat lunak dan perancangan perangkat keras. Pada bagian implementasi sistem terdiri dari implementasi perangkat keras dan implementasi perangkat lunak.

#### 5.1.2 Perancangan Perangkat Keras

Perancangan perangkat keras sistem klasifikasi frekuensi penggunaan minyak goreng dapat dilihat pada Gambar 5.2. Sistem menggunakan sensor warna TCS3200 dan sensor fotodioda. Mikrokontroler yang digunakan adalah Arduino Uno, serta LCD yang digunakan adalah LCD 1602A ukuran 16x2. Sensor warna TCS3200 dan Sensor fotodioda disambungkan ke mikrokontroler Arduino Uno. Kemudian, data yang telah didapat dari sensor akan diproses dengan metode *Support Vector Machine* pada *python*. Setelah selesai diproses oleh *python*, hasil klasifikasi frekuensi penggunaan minyak goreng menggunakan metode *Support Vector Machine* akan ditampilkan pada LCD 16x2.

Peletakan posisi LCD16x2 berada didepan *prototype* dan posisi sensor warna TCS3200 diatas gelas uji dengan jarak 2 cm, sensor *photodiode* berada diposisi bawah gelas penguji minyak goreng dengan memanfaatkan cahaya yang keluar dari sensor warna TCS3200 sebagai *input photodiode*. Kemudian minyak yang terdapat pada gelas uji harus sesuai dengan tinggi label yaitu 100 ml. Peletakan

sensor tersebut bertujuan supaya sensor dapat mendeteksi warna dan tingkat kekeruhan dari minyak goreng secara akurat. LCD 16x2 didepan *proTOTYPE* untuk memudahkan melihat langsung hasil klasifikasi. Berikut ini merupakan gambaran rangkaian dalam sistem yang ditunjukkan pada Gambar 5.3 dan penggunaan pin mode pada tabel 5.1.



**Gambar 5.3 Skematik Perancangan Perangkat Keras**

**Tabel 5.1 Koneksi PIN Perangkat Keras**

Pin Arduino Uno	Pin Sensor Warna	Pin Sensor Photodiode	Pin LCD 16x2 I2C
VCC	VCC	VCC	VCC
GND	GND	GND	GND
A1	-	-	-
A0	-	A0	-
A4	-	-	A4
A5	-	-	A5
Pin 3	Pin 3	-	-
Pin 4	Pin 4	-	-
Pin 5	Pin 5	-	-
Pin 6	Pin 6	-	-

Pin 7	Pin 7	-	-
-------	-------	---	---

### 5.1.3 Perancangan Perangkat Lunak

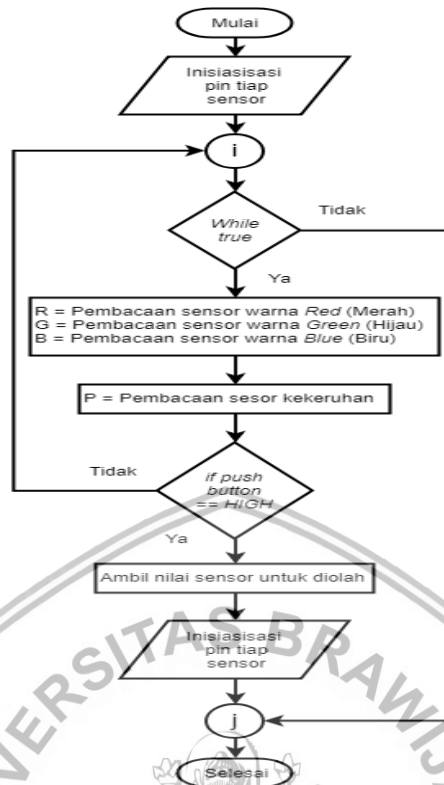
Perangkat lunak yang digunakan dalam pembuatan sistem ini adalah Arduino IDE. Program yang dibuat dengan menggunakan Bahasa pemrograman *python* adalah metode *Support Vector Machine* sebagai pengambilan keputusan untuk menentukan penggunaan minyak goreng dengan objek penggorengan yang sudah ditentukan berdasarkan input dari sensor TCS3200 dan sensor fotodiode. Perancangan perangkat lunak dimulai dari input masing-masing sensor yaitu sensor warna TCS3200 dan sensor fotodiode..

#### 5.1.3.1 Perancangan Pengambilan Data Latih

Sistem diawali dengan inisialisasi sensor warna TCS3200 dan sensor fotodiode dengan Input berupa warna R,G,B dan kekeruhan dari minyak goreng. Dari inputan tersebut didapatkan data latih tentang kelayakan minyak goreng dari penggorengan 1 sampai 3 kali dengan objek yang berbeda yaitu ikan dan non-ikan. Pengklasifikasian awal dari sistem berupa pemisahan kelas penggorengan ikan dan non-ikan untuk mengetahui kondisi awal minyak goreng. Kemudian dari hasil klasifikasi ikan dan non ikan dilakukan lagi pengklasifikasian berupa kelayakan minyak goreng di ukur dari berapa kali minyak goreng tersebut sudah digunakan dengan mengukur warna dan tingkat kekeruhan minyak goreng. Batas yang digunakan untuk menentukan minyak tersebut layak atau tidaknya adalah 3x penggorengan (Nora, 2016). Dari batas yang sudah ditentukan tersebut dapat ditemukan pembagian kelas minyak goreng termasuk kedalam kelas masih layak digunakan atau tidak. Berikut ini merupakan hasil pengambilan data latih dari tiap-tiap kelas yang ditunjukkan pada Lampiran dan flowchart utama pada Gambar 5.4.

Merujuk pada Gambar 5.4 ketika *user* menekan *Pushbutton* maka nilai tersebut diolah dan dilakukan klasifikasi menggunakan metode SVM. Hasil dari perhitungan berupa klasifikasi kelayakan minyak goreng, termasuk kedalam kelas masih layak atau tidak untuk digunakan kembali. Jika kondisi minyak goreng masih layak atau tidak, maka LCD akan menampilkan tulisan “Layak” atau “Tidak Layak”.

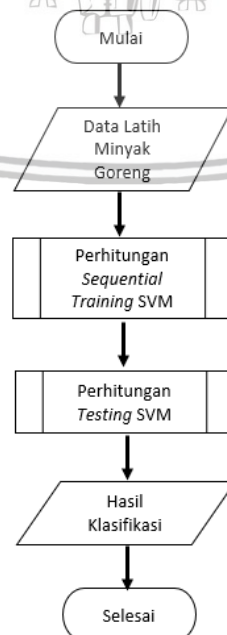
Pada penerapan implementasi metode klasifikasi *Support Vector Machine* (SVM) terhadap pemakaian minyak goreng yang akan diteliti, maka akan dibuat sebuah variabel *array* yang digunakan sebagai fitur input yang berasal dari nilai kalibrasi sensor TCS3200 dan juga nilai kalibrasi dari sensor *photodiode*. Pada sensor TCS3200 melakukan proses *mapping* menggunakan kertas warna putih dan hitam untuk dapat menentukan warna *Red*, *Green*, dan *Blue* sudah sesuai dengan nilai RGB pada *Coreldraw photoPaint*. Jika sesuai dengan hasil *photoPaint* maka dapat dilakukan proses pengambilan data latih untuk sensor TCS3200. Sedangkan pada sensor *photodiode*, kalibrasi dilakukan dengan melakukan perhitungan pada nilai *analog A0* yang kemudian diolah dengan ketentuan perhitungan pada *photodiode* untuk menentukan tingkat kekeruhan sesuai dengan intensitas cahaya yang didapatkan oleh sensor.



Gambar 5.4 Perancangan Pengambilan Data Latih

### 5.1.3.2 Perancangan Algoritma *Support Vector Machine* (SVM)

Langkah pertama yang dilakukan dalam proses SVM yaitu dengan mengambil data latih, kemudian tahap selanjutnya perhitungan dengan *kernel* yang digunakan yaitu *kernel linier* yang akan menghasilkan matrik *Hessian*. Gambar 5.5 merupakan tahapan pada proses SVM.



Gambar 5.5 Proses *Support Vector Machine*

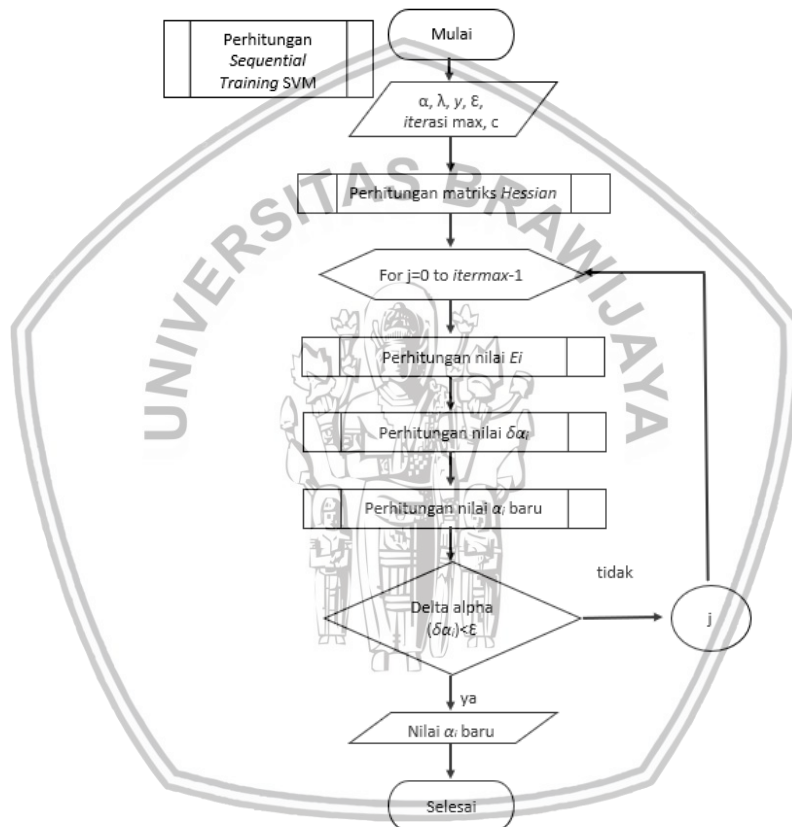


Penjelasan :

- Memasukkan data latih.
- Melakukan perhitungan sequential training SVM.
- Melakukan perhitungan *testing* SVM.
- Hasil klasifikasi, merupakan kelas prediksi yang dihasilkan dari data uji.

### 5.1.3.3 Perhitungan Sequential Training SVM

Tahap *sequential training* SVM dilakukan agar mendapatkan *hyperplane* yang optimal. Gambar 5.6 merupakan tahapan *sequential training* SVM.



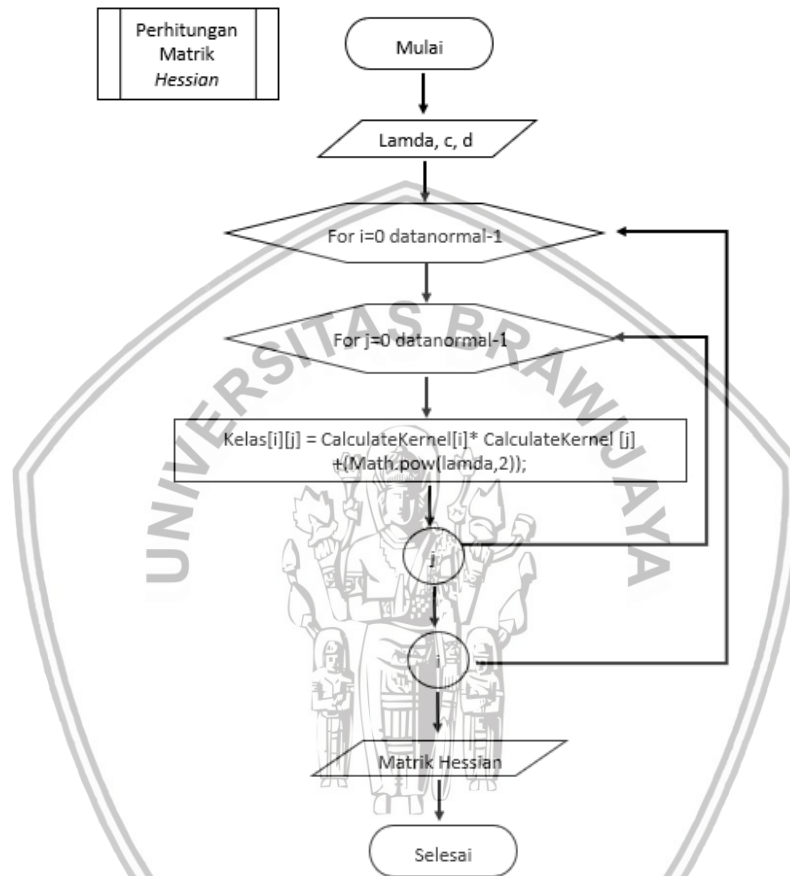
**Gambar 5.6 Perhitungan Sequential Training SVM**

Berikut ini merupakan penjelasan dari perhitungan *sequential training* diatas.

- Inisialisasi nilai parameter SVM yaitu  $\alpha_i$ ,  $\lambda$ ,  $\gamma$ ,  $\epsilon$ , *iterasi* max,  $C$  dan fungsi *kernel* yang digunakan dalam penelitian ini adalah *kernel Linier*.
- Menghitung nilai matrik *Hessian* sesuai dengan persamaan 2.8.
- Menghitung nilai  $E_i$  dengan menggunakan persamaan 2.9.
- Kemudian mencari nilai  $\delta\alpha_i$  sesuai dengan persamaan 2.10.
- Setelah nilai  $\delta\alpha_i$  didapatkan, selanjutnya melakukan iterasi sejumlah *itermax* yang sudah ditentukan.
- Langkah berikutnya menghitung nilai  $\alpha_i$  sesuai dengan persamaan 2.11.

7. Iterasi dilakukan jika  $(|\delta\alpha_i|) < \varepsilon$  atau telah mencapai sejumlah iterasi maksimal yang sudah ditentukan. Proses ini dilakukan sampai mendapatkan nilai dari  $\alpha_i$  mengalami perubahan.
8. Hasil keluaran berupa nilai  $\alpha_i$  baru yang disebut *hyperplane* yang optimal dan akan digunakan dalam proses *testing SVM*.

#### 5.1.3.4 Perhitungan Matriks Hessian



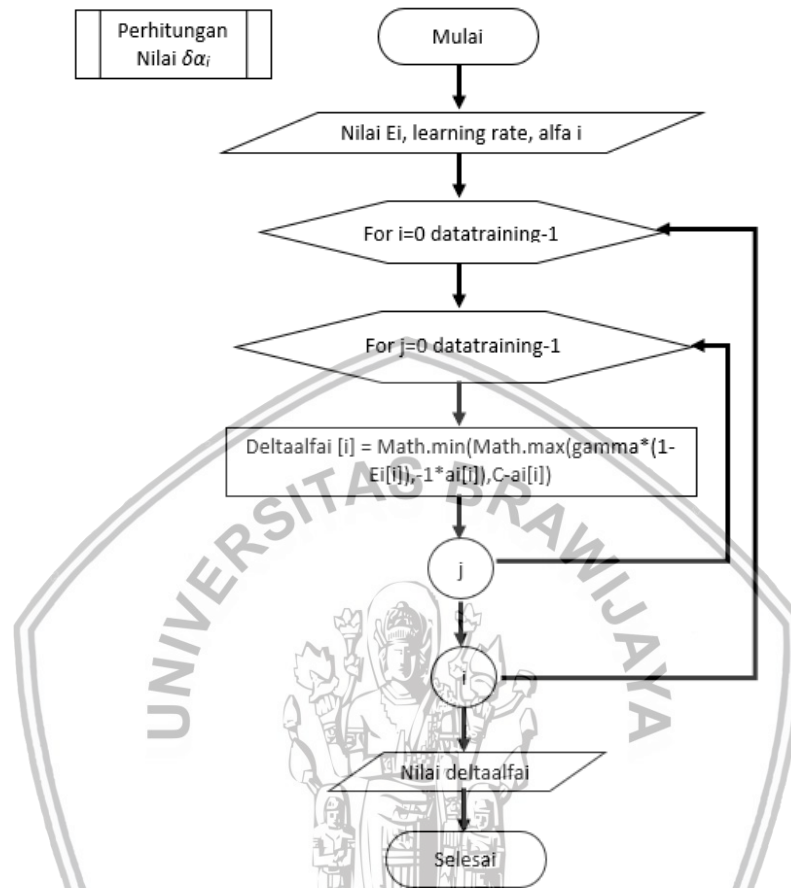
**Gambar 5.7 Perhitungan Matriks Hessian**

Penjelasan tentang tahapan perhitungan matriks *hessian* pada Gambar 5.7 sebagai berikut.

1. Memasukkan sejumlah data latih.
2. Inisialisasi nilai parameter  $\lambda$ ,  $C$  dan  $d$  untuk perhitungan *kernel polynomial of degree 2*.
3. Menghitung matrik *Hessian* sesuai dengan persamaan 2.8.
4. Perulangan untuk sejumlah data latih.
5. Hasil keluaran berupa matrik Hessian

### 5.1.3.5 Perhitungan Nilai $\delta\alpha_i$

Langkah perhitungan nilai  $\delta\alpha_i$ , dapat dilihat pada Gambar 5.8 dibawah ini.



**Gambar 5.8 Perhitungan Nilai  $\delta\alpha_i$**

Penjelasan tentang tahapan perhitungan nilai  $\delta\alpha_i$  sebagai berikut.

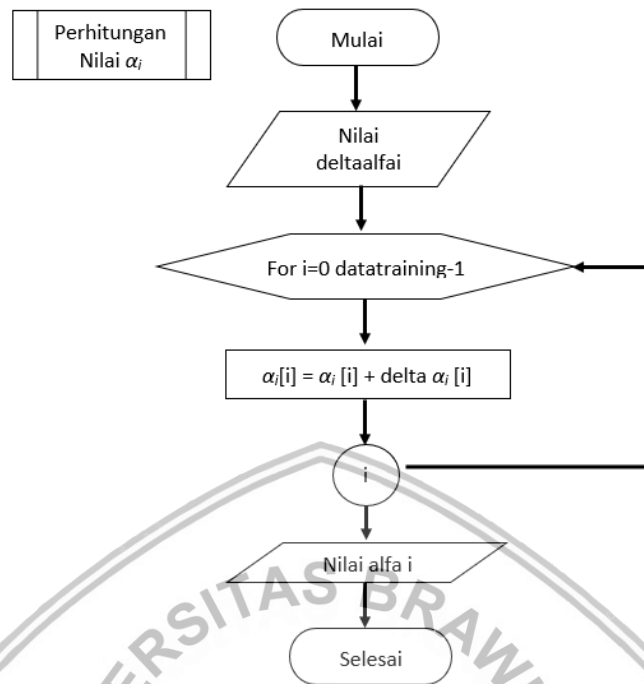
1. Memasukkan nilai matriks  $Ei$  yang sudah dihitung sebelumnya, *learning rate*, dan  $\alpha_i$ .
2. Menghitung nilai  $\delta\alpha_i$ , sesuai dengan persamaan 2.10.
3. Melakukan perulangan sesuai dengan panjang data latih.
4. Hasil keluaran berupa nilai  $\delta\alpha_i$ .

### 5.1.3.6 Perhitungan Nilai $\alpha_i$

Berikut ini merupakan penjelasan dari tahapan perhitungan nilai  $\alpha_i$ .

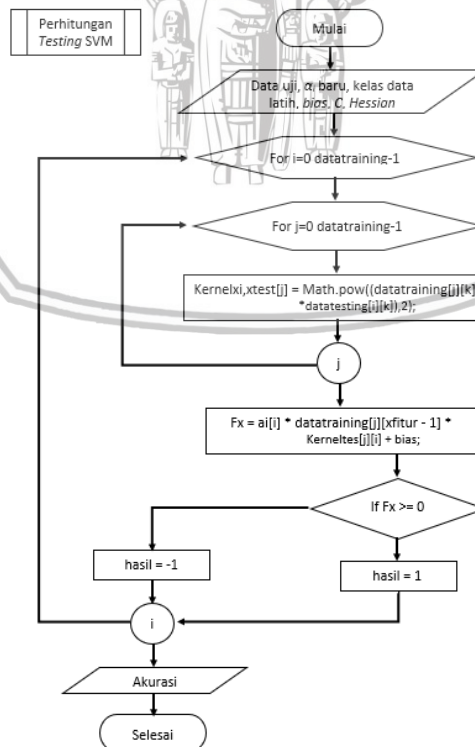
1. Memasukkan nilai  $\delta\alpha_i$ .
2. Melakukan perulangan sesuai dengan panjang data latih.
3. Menghitung nilai  $\alpha_i$ , sesuai dengan persamaan 2.11.
4. Hasil keluaran berupa nilai  $\alpha_i$ .

Langkah perhitungan nilai  $\alpha_i$  dapat dilihat pada Gambar 5.9 dibawah ini.



Gambar 5.9 Perhitungan Nilai  $\alpha_i$

### 5.1.3.7 Proses Perhitungan Testing SVM



Gambar 5.10 Perhitungan Testing SVM

Berikut ini Penjelasan tahapan-tahapan perhitungan *testing* SVM dari Gambar 5.10.

1. Memasukkan nilai data uji, nilai  $\alpha_i$ , *bias*, dan  $y$ .
2. Melakukan perulangan sesuai dengan panjang data uji.
3. Melakukan perhitungan nilai  $K(x_i, x_j)$  pada data uji.
4. Melakukan perhitungan nilai  $f(x) = D_{ij} K(x_i, x_j)$ .
5. Hasil keluaran merupakan nilai hasil dari klasifikasi.

### 5.1.3.8 Perhitungan Manual Sequential Training SVM

Perhitungan manual penelitian ini, menggunakan dataset yang berupa nilai dari sensor TCS3200 yaitu nilai R, G, B dengan masing-masing nilai sesuai dengan nilai pembacaan sensor yang sudah di kalibrasi dengan warna putih. Kemudian nilai P dari inputan sensor fotodiode yang juga sudah dikalibrasi dengan nilai *voltase* Arduino Uno dengan *range* nilai 0 sampai 5 *volt*, untuk kondisi cahaya dapat diterima langsung oleh sensor dan juga adanya penghalang cahaya menuju ke sensor. Data latih yang digunakan dalam perhitungan manual ini menggunakan 5 data latih, dan 2 data uji. Dengan masing-masing kelas dari bekas minyak yang digunakan menggoreng non ikan dengan nilai 1, dan kelas minyak bekas penggorengan ikan dengan nilai -1.

Proses perhitungan sequential training SVM dilakukan untuk mengklasifikasi data dari beberapa objek, yang mana memiliki *hyperplane* atau bidang pemisah antara 2 objek dari 2 kelas yang memiliki fungsi pemisah dengan nilai margin yang maksimal. Data latih yang akan digunakan pada penelitian ini terdapat pada table dibawah, dengan data ke 1-sampai dengan data ke-5. Pada baris ke-1 dan ke-2 merupakan data minyak hasil penggorengan non ikan dengan kelas 1. Kemudian baris ke-3 sampai ke-5 merupakan data minyak hasil penggorengan ikan dengan kelas -1.

Tabel 5.2 Data Latih

No.	R	G	B	P	Kelas	Y
1	90	18	61	0,22	Layak	1
2	90	18	61	0,22	Layak	1
3	21	79	40	4,26	Tidak Layak	-1
4	21	79	40	4,26	Tidak Layak	-1
5	21	79	40	4,26	Tidak Layak	-1

Proses perhitungan *sequential training* pada metode SVM adalah sebagai.

1. Inisialisasi  $a_i = 0$ ,  $\gamma = 0,000001$ ,  $C = 1$ ,  $\lambda = 0,5$ .
2. Menghitung nilai matrik Hessian seperti pada persamaan 2.8, yaitu
 
$$D_{ij} = y_i y_j (K(x_i, x_j) + \lambda^2)$$
,
 yang mana  $y_i$  = kelas atribut pertama, dan  $y_j$  = kelas pada atribut ke 2.  $K(x_i, x_j)$  = merupakan fungsi *kernel*



yang digunakan untuk menyelesaikan perhitungan dengan menggunakan fungsi *kernel Linier* pada persamaan 2.4. Seperti contoh dibawah ini dari perhitungan data ke 1 sampai ke 3.

$$\begin{aligned}
 D_{11} &= y_1 y_1 (K(x_1, x_1) + \lambda^2) \\
 &= 1 * 1(((90 * 90) + (18 * 18) + (61 * 61) + (0,22 * 0,22)) + 0,25) \\
 &= 12145 \\
 D_{12} &= y_1 y_2 (K(x_1, x_2) + \lambda^2) \\
 &= 1 * 1(((90 * 90) + (18 * 18) + (61 * 61) + (0,22 * 0,22)) + 0,25) \\
 &= 12145 \\
 D_{13} &= y_1 y_3 (K(x_1, x_3) + \lambda^2) \\
 &= 1 * -1(((90 * 21) + (18 * 79) + (61 * 40) + (0,22 * 4,26)) + 0,25) \\
 &= -5753
 \end{aligned}$$

**Tabel 5.3 Perhitungan Matriks Hessian**

$D_{ij}$	1	2	3	4	5
1	12145	12145	-5753	-5753	-5753
2	12145	12145	-5753	-5753	-5753
3	-5753	-5753	8300	8300	8300
4	-5753	-5753	8300	8300	8300
5	-5753	-5753	8300	8300	8300

3. Langkah selanjutnya setelah mendapatkan hasil matriks *Hessian* yaitu menghitung nilai *error*, menggunakan persamaan 2.9 sampai persamaan 2.11

- a. Menghitung nilai  $E_i$  menggunakan persamaan 2.9  $E_i = \sum_{j=1}^n a_i D_{ij}$  dengan inisialisasi awal  $a_i = 0$ .

Berikut merupakan contoh perhitungan  $E_i$  untuk data ke-1 dan data ke-2

$$\begin{aligned}
 E_1 &= (0 * 12145) + (0 * 12145) + (0 * (-5753)) + (0 * (-5753)) \\
 &\quad + (0 * (-5753)) = 0
 \end{aligned}$$

$$\begin{aligned}
 E_2 &= (0 * 12145) + (0 * 12145) + (0 * (-5753)) + (0 * (-5753)) \\
 &\quad + (0 * (-5753)) = 0
 \end{aligned}$$

Perhitungan nilai  $E_i$  dilakukan untuk semua data latih, hasil dari perhitungan tersebut dapat dilihat pada table di bawah ini.

**Tabel 5.4 Perhitungan Nilai  $E_i$  pada Iterasi ke 1**

$I$	$E_i$
1	0
2	0
3	0
4	0
5	0

- b. Menghitung nilai  $\delta a_i$  dengan menggunakan persamaan 2.10. berikut contoh perhitungan manual untuk data ke-1.

$$\begin{aligned}
 \delta a_1 &= \min(\max[\gamma(1 - E_1), a_1], C - a_1) \\
 &= \min(\max[0,000001(1 - 0), 0], 1 - 0) \\
 &= \min(\max[0,000001(1), 0], 1) \\
 &= \min(\max[0,000001], 1) = 0,01
 \end{aligned}$$

Hasil dari perhitungan seluruh data dapat dilihat pada table di bawah ini.

**Tabel 5.5 Perhitungan Nilai  $\delta a_1$  pada Iterasi ke 1**

$I$	$\delta a_1$
1	0,000001
2	0,000001
3	0,000001
4	0,000001
5	0,000001

- c. Tahap selanjutnya memperbarui nilai  $a_i$ , dapat dihitung dengan persamaan 2.11. Berikut contoh perhitungan manual untuk data ke-1.

$$\begin{aligned}
 a_1 &= a_1 + \delta a_1 \\
 &= 0 + 0,000001 \\
 &= 0,000001
 \end{aligned}$$

Hasil dari perhitungan seluruh data dapat dilihat pada table dibawah ini.

**Tabel 5.6 Perhitungan Nilai  $a_1$  pada Iterasi ke 1**

$l$	$a_i$
1	0,000001
2	0,000001
3	0,000001
4	0,000001
5	0,000001

4. Melakukan perhitungan Iterasi untuk setiap  $i = 1, 2, 3, \dots, n$ . Proses perhitungan pada perhitungan manualisasi dilakukan Iterasi (*itermax*) sebanyak 2 kali. Tahapan iterasi dimulai dengan menghitung nilai *error*.

Masuk Iterasi ke -2

- a. Menghitung nilai  $E_i$  dengan persamaan 2.9 dan inisialisasi  $a_i$  baru yaitu pada table 5.6. Hasil perhitungan seluruh data pada iterasi ke-2 dapat dilihat pada table dibawah ini.

**Tabel 5.7 Perhitungan nilai  $E_i$  pada Iterasi ke 2**

$l$	$E_i$
1	0,070
2	0,070
3	0,134
4	0,134
5	0,134

- b. Tahapan selanjutnya menghitung nilai  $\delta a_i$  untuk iterasi ke-2 dengan menggunakan persamaan 2.10. Hasil perhitungan seluruh data dapat dilihat pada table dibawah ini.

**Tabel 5.8 Perhitungan Nilai  $\delta a_i$  pada Iterasi ke 2**

$l$	$\delta a_i$
1	0,000000930
2	0,000000930
3	0,000000866
4	0,000000866
5	0,000000866

- c. Memperbarui nilai  $a_i$  dengan menggunakan persamaan 2.11, yang mana nilai  $a_i$  menggunakan nilai baru yaitu pada table 5.6. Hasil dari perhitungan seluruh data dapat dilihat pada table dibawah ini.

**Tabel 5.9 Perhitungan Nilai  $a_i$  pada Iterasi ke 2**

$I$	$a_i$
1	0,000001930
2	0,000001930
3	0,000001866
4	0,000001866
5	0,000001866

### 5.1.3.9 Perhitungan Manual Testing SVM

Proses selanjutnya setelah melakukan proses *training* melakukan proses *testing* untuk data uji. Pada proses ini dilakukan untuk mendapatkan nilai  $f(x)$  yang mana dapat menggunakan persamaan 2.12. Data uji yang digunakan dalam proses perhitungan manual dapat dilihat pada table 5.10, yang mana terdapat 4 inputan yaitu RGB sebagai kriteria masukan warna yang terdapat pada minyak goreng, dan juga Photodiode sebagai kriteria masukan kekeruhan dari minyak goreng yang di teliti. Parameter  $y$  merupakan inisialisasi untuk nilai 1 merupakan kelas penggorengan non ikan, sedangkan nilai -1 merupakan kelas penggorengan ikan.

**Tabel 5.10 Data Uji**

No.	R	G	B	P	Kelas	Y
6	90	18	61	0,22	Layak	1
7	21	79	40	4,26	Tidak Layak	-1

Untuk mencari nilai  $f(x)$  sebelumnya mencari nilai *bias* yang mana merupakan posisi bidang relatif terhadap pusat koordinat. Nilai  $K(x_i, x^+)$  merupakan nilai *support vector* kelas yang paling besar, sedangkan nilai  $K(x_i, x^-)$  merupakan nilai *support vector* kelas yang paling kecil.

Contoh perhitungan manual untuk  $K(x_i, x^+)$  didapatkan berdasarkan nilai  $a_i$  terbesar dari kelas positif yakni 0,000001930, sedangkan nilai  $K(x_i, x^-)$  didapatkan berdasarkan nilai  $a_i$  yang paling besar dari kelas negatif yaitu 0,000001866. Berdasarkan nilai tersebut dilihat dari nilai matriks *Hessian* kolom 1 dan kolom 3, yang mana dapat dilihat pada table dibawah ini.

**Tabel 5.11 Nilai Matriks *Hessian***

$D_{ij}$	1	3
1	12145	-5753
2	12145	-5753
3	-5753	8300
4	-5753	8300
5	-5753	8300

Langkah selanjutnya yaitu menghitung nilai  $K(x_i, x^+)$  dan nilai  $K(x_i, x^-)$  dengan menggunakan persamaan 2.14. Berikut contoh perhitungan manual.

$$\begin{aligned}
 K(x_i, x^+) &= \sum_{i=1}^5 a_i y_i K(x_i, x^+) \\
 &= (0,000001930 * 1 * 12145) + (0,000001930 * 1 * 12145) \\
 &\quad + (0,000001866 * (-1) * (-5753)) \\
 &\quad + (0,000001866 * (-1) * (-5753)) \\
 &\quad + (0,000001866 * (-1) * (-5753)) \\
 &= 0,07908055 \\
 K(x_i, x^-) &= \sum_{i=1}^5 a_i y_i K(x_i, x^-) \\
 &= (0,000001930 * 1 * (-5753)) + (0,000001930 * 1 * (-5753)) \\
 &\quad + (0,000001866 * (-1) * 8300) \\
 &\quad + (0,000001866 * (-1) * 8300) \\
 &\quad + (0,000001866 * (-1) * 8300) \\
 &= -0,06867065
 \end{aligned}$$

Berikut nilai  $K(x_i, x^+)$  dan nilai  $K(x_i, x^-)$ , dapat dilihat pada table dibawah ini.

**Tabel 5.12 Nilai  $X^+$  dan  $X^-$**

$x^+$	0,07908055
$x^-$	-0,06867065

Nilai  $K(x_i, x^+)$  dan nilai  $K(x_i, x^-)$  sudah dihitung, kemudian langkah selanjutnya yaitu menghitung nilai *bias* dengan menggunakan persamaan 2.13. Berikut contoh perhitungan manual.

$$b = \frac{1}{2} \left[ \sum_{i=1}^5 a_i y_i K(x_i, x^+) + \sum_{i=1}^5 a_i y_i K(x_i, x^-) \right]$$



$$= \frac{1}{2} [0,07908055 + (-0,06867065)]$$

$$= -0,005204951$$

Setelah didapatkan nilai *bias*, maka proses selanjutnya yaitu menghitung nilai  $f(x)$  dengan menggunakan persamaan 2.12. Berikut merupakan contoh perhitungan manual.

Tahap awal mencari nilai  $K(x_i, x_{test1})$  untuk data uji ke-1

$$\sum_{i=1}^5 y_i y_j K(x_i, x) = (90 * 90) + (18 * 18) + (61 * 61) + (0,22 * 0,22)$$

$$= 12145$$

Tahap selanjutnya mencari nilai  $K(x_i, x_{test2})$  untuk data uji ke-1

$$\sum_{i=1}^5 y_i y_j K(x_i, x) = (90 * 21) + (18 * 79) + (61 * 40) + (0,22 * 4,26)$$

$$= 5753$$

Hasil nilai  $K(x_i, x_{test1})$  dan nilai  $K(x_i, x_{test2})$  untuk seluruh data dapat dilihat pada table dibawah ini.

**Tabel 5.13 Perhitungan Nilai  $K(x_i, x_{test1})$  dan nilai  $K(x_i, x_{test2})$**

$K(x_i, x_{testj})$	$x_{test1}$	$x_{test2}$
x1	12145	5753
x2	12145	5753
x3	5753	8300
x4	5753	8300
x5	5753	8300

Setelah nilai  $K(x_i, x_{testj})$  sudah didapatkan, langkah selanjutnya yaitu mencari nilai  $f(x)$

$$f(x_{test1}) = \sum_{i=1}^5 a_i y_j K(x_i, x_{test1}) + b$$

$$= ((0,000001930 * 1 * 12145) + (0,000001930 * 1 * 12145) + (0,000001866 * (-1) * 5753) + (0,000001866 * (-1) * 5753) + (0,000001866 * (-1) * 5753)) + (-0,005204951)$$

$$= 0,00946156$$

$$\begin{aligned}
 f(x_{test2}) &= \sum_{i=1}^5 a_i y_i K(x_i, x_{test2}) + b \\
 &= (0,000001930 * 1 * 5753) + (0,000001930 * 1 * 5753) \\
 &\quad + (0,000001866 * (-1) * 8300) \\
 &\quad + (0,000001866 * (-1) * 8300) \\
 &\quad + (0,000001866 * (-1) * 8300) + (-0,005204951) \\
 &= -0,02946770
 \end{aligned}$$

Setelah mendapatkan nilai  $f(x)$  masing-masing kelas, maka sudah dapat diklasifikasikan berdasarkan hasil dari nilai  $f(x)$  tersebut. Nilai  $> 0$  menjelaskan bahwa data tersebut masuk kedalam kategori kelas 1 yaitu layak, sedangkan nilai  $< 0$  menjelaskan bahwa data tersebut termasuk dalam kategori kelas -1 yaitu tidak layak. Hasil klasifikasi data uji dapat dilihat pada table dibawah ini.

**Tabel 5.14 Hasil Klasifikasi**

	Kelas prediksi	Kelas hasil
Fxtest1 0,00946156	1	1
Fxtest2 -0,02946770	-1	-1

## 5.2 Implementasi Sistem

Implementasi sistem merupakan tahap untuk merealisasikan pembuatan sistem berdasarkan semua perancangan yang telah dilakukan sebelumnya. Pada subbab ini menjelaskan satu per satu secara rinci terkait implementasi prototype, implementasi perangkat keras serta implementasi perangkat lunak.

### 5.2.1 Implementasi Perangkat Keras

Implementasi algoritma SVM untuk klasifikasi kelayakan minyak goreng berdasarkan warna dan tingkat kekeruhan menggunakan perangkat keras yang dapat dilihat pada table dibawah ini.

**Tabel 5.15 Spesifikasi Perangkat Keras**

Komponen	Spesifikasi
Prosesor	Intel Core i5-8250U
Memory	4 GB
Hardisk	500 GB

Selain menggunakan laptop dengan spesifikasi yang disebutkan diatas. Sistem ini juga menggunakan beberapa komponen yaitu mikrokontroler, sensor dan LCD 16x2. Mikrokontroller yang digunakan adalah Arduino Uno, kemudian sensor warna yang digunakan adalah TCS3200, dan sensor Photodioda untuk

kekeruhan dari minyak goreng. Sedangkan untuk outputnya akan ditampilkan pada LCD 16x2 yang ditunjukkan pada gambar dibawah ini.



**Gambar 5.11 *Prototype* Alat Pendeteksi Kelayakan Minyak Goreng**

Pada gambar 5.10 menunjukkan implementasi perangkat keras yang berbentuk tabung dan terbuat dari bahan plastik. Sensor warna, sensor photodiode, dan mikrokontroler Arduino Uno berada didalam tabung. Serta LCD yang berada didepan untuk memudahkan dalam memonitoring proses dan juga hasil dari klasifikasi kelayakan minyak goreng.



**Gambar 5.12 Tempat Peletakan Gelas Uji Minyak Goreng**

Pada gambar 5.11 menunjukkan interior dalam prototype untuk meletakkan gelas uji dengan sensor TCS 3200 berada dibagian atas gelas uji. Kemudian sensor Photodiode berada dibagian bawah gelas uji, hal ini dimaksudkan agar cahaya

yang memancar dari sensor TCS 3200 dapat diterima langsung oleh sensor Photodiode.

## 5.2.2 Implementasi Perangkat Lunak

Implementasi algoritma SVM untuk klasifikasi kelayakan minyak goreng berdasarkan warna dan tingkat kekeruhan menggunakan perangkat lunak yang dapat dilihat pada table dibawah ini.

**Tabel 5.16 Spesifikasi Perangkat Lunak**

Komponen	Spesifikasi
Sistem operasi	Windows 10
Bahasa pemrograman	C dan Python
<i>Tools</i> pemrograman	Arduino IDE dan Visual Studio Code

Pada bagian implementasi ini terdapat beberapa bagian yaitu program pembacaan sensor, perhitungan algoritma SVM, melakukan inferensi penalaran dan perhitungan matematika yang cukup kompleks sebagai fungsi untuk menentukan output sistem. Program ini menggunakan Bahasa C pada Arduino IDE dan Python pada Visual Studio Code sebagai pemroses perhitungan SVM.

### 5.2.2.1 Pembacaan Nilai Sensor Warna TCS 3200

Implementasi *source code* terhadap inisialisasi pin sensor TCS 3200 sebagai pemroses warna minyak goreng berdasarkan nilai R, G, B yang sudah di kalibrasi nilai perhitungan awal sensor TCS 3200 dengan menggunakan kertas putih dan juga kertas hitam sebagai pengakurasi nilai warna yang akan di proses pada sensor untuk didapatkan nilainya. Berikut ini merupakan *code* penginisialisasian sensor warna TCS 3200.

**Tabel 5.17 Kode Program Sensor Warna TCS 3200**

Baris	Source Code
1	for (int i = 0; i < 10; i++) {
2	digitalWrite(S2, LOW);
3	digitalWrite(S3, LOW);
4	// Reading the output frequency
5	frequency = pulseIn(sensorOut, LOW);
6	R = frequency;
7	ratarataMerah += R;
8	delay(25);
9	
10	// Setting Green filtered photodiodes to be read
11	digitalWrite(S2, HIGH);
12	digitalWrite(S3, HIGH);
13	// Reading the output frequency
14	frequency = pulseIn(sensorOut, LOW);
15	G = frequency;
16	ratarataHijau += G;
17	delay(25);
18	
19	// Setting Blue filtered photodiodes to be read
20	digitalWrite(S2, LOW);

21	<code>digitalWrite(S3, HIGH);</code>
22	<code>// Reading the output frequency</code>
23	<code>frequency = pulseIn(sensorOut, LOW);</code>
24	<code>B = frequency;</code>
25	<code>ratarataBiru += B;</code>
26	<code>delay(25);</code>
27	<code>}</code>
28	
29	<code>//pencarian rata-rata</code>
30	<code>ratarataMerah = ratarataMerah / 10;</code>
31	<code>ratarataHijau = ratarataHijau / 10;</code>
32	<code>ratarataBiru = ratarataBiru / 10;</code>
33	
34	<code>ratarataMerah = map(ratarataMerah, 94,468,255,0);</code>
35	<code>ratarataHijau = map(ratarataHijau, 98,640,255,0);</code>
36	<code>ratarataBiru = map(ratarataBiru, 82,473,255,0);</code>

### 5.2.2.2 Pembacaan Nilai Sensor Kekeruhan Photodiode

Implementasi *source code* terhadap inisialisasi pin sensor fotodiode pada port A0 dan pembacaan nilai sensor fotodiode dalam mendeteksi tingkat kekeruhan minyak yang ditunjukkan pada table di bawah ini.

**Tabel 5.18 Kode Program Sensor *Photodiode***

Baris	Source Code
1	<code>for (int i = 0; i &lt; 10; i++){</code>
2	<code>int reading = analogRead(A0);</code>
3	
4	<code>Tegangan += reading * 5 / 1023.0;</code>
5	<code>}</code>
6	<code>Tegangan = Tegangan/10;</code>

### 5.2.2.3 Pengiriman Data Menggunakan Simple Json

Pengiriman data dari Arduino menuju ke python menggunakan *library* *Arduinojson* pada Arduino IDE dan juga *simpeljson* pada import *python*. Nilai dari sensor yang sudah didapat kemudian dimasukkan kedalam input data pada *python* dengan nilai *array* berupa r, g, b, p. Berikut ini merupakan kode program *Json* pada Arduino IDE dan juga *python*.

**Tabel 5.19 Kode Program Arduino *Json***

Baris	Source Code
1	<code>StaticJsonBuffer&lt;200&gt; jsonBuffer;</code>
2	
3	<code>JsonObject&amp; root = jsonBuffer.createObject();</code>
4	
5	<code>root["r"] = ratarataMerah;</code>
6	<code>root["g"] = ratarataHijau;</code>
7	<code>root["b"] = ratarataBiru;</code>
8	<code>root["p"] = Tegangan;</code>
9	
10	<code>Serial.print("[ ]\n");</code>



**Tabel 5.20 Kode Program *Simple Json***

Baris	Source Code
1	while 1:
2	arduinoData = ser.readline()
3	jsonData = json.loads(arduinoData)
4	if jsonData:
5	break
6	for q in fileData:
7	t0 = time.time()
8	benar = 0
9	tester = loader.loadData(q, 1)
10	for t in tester:
11	t =
12	[jsonData['r'],jsonData['g'],jsonData['b'],jsonData['p']]

#### 5.2.2.4 Algoritma Perhitungan Matriks *Hessian*

Perhitungan matriks *Hessian* dilakukan untuk mencari nilai *Hessian* yang nantinya akan digunakan pada proses *sequential training SVM*. Hasil tersebut didapatkan dari nilai *kernel* dan *lambda*. Kode program pada perhitungan matriks *Hessian* dapat dilihat pada table dibawah ini.

**Tabel 5.21 Proses Matrik *Hessian***

Baris	Source Code
1	def setMaxMinData(self, maxD, minD):
2	self.normalMax = maxD
3	self.normalMin = minD
4	# dipanggil dulu *
5	
6	def generateMinMaxData(self):
7	self.normalMax =
8	self.generateNormal(len(self.data[0])-1, -99999999)
9	self.normalMin =
10	self.generateNormal(len(self.data[0])-1, 99999999)
11	for i in self.data:
12	for j in range(len(i)-1):
13	if self.normalMax[j] < i[j]:
14	self.normalMax[j] = i[j]
15	if self.normalMin[j] > i[j]:
16	self.normalMin[j] = i[j]
17	# print(self.normalMax)
18	# print(self.normalMin)
19	
20	# dipanggil dulu
21	def normalizeData(self):
22	self.dataNormal = []
23	for i in range(len(self.data)):
24	temp = []
25	for j in range(len(self.data[i])-1):
26	temp.append((self.data[i][j] -
27	self.normalMin[j]) /
28	((self.normalMax[j] -
29	self.normalMin[j])*1.0))
30	# temp.append(self.data[i][j])
31	self.dataNormal.append(temp)
32	# pprint.pprint(self.dataNormal, width=500)
33	# dipanggil dulu
34	
35	def generateAlpha(self):

36	self.alphaList = []
37	self.deltaAlpha = []
38	self.Error = []
39	for i in range(len(self.dataNormal)):
40	self.alphaList.append(self.alpha)
41	self.deltaAlpha.append(1)
42	self.Error.append(0)
43	
44	def CalculateKernel(self, x, y):
45	# print("calculate kernel ")
46	# print(x, y)
47	result = 0
48	for idx in range(len(x)): # -----
49	result += x[idx]*y[idx]
50	result = math.pow(result+self.constant, 2)
51	return result
52	
53	def CalculateHessian(self, kelasX, kelasY, x, y):
54	return kelasX*kelasY*(self.CalculateKernel(x,
55	y)+math.pow(self.lambd, 2))

Tabel 5.17 diatas menunjukkan kode program yang menjelaskan perhitungan matriks Hessian yaitu pada baris 60 sampai 63 dengan memperhitungkan nilai masukkan terlebih dahulu dan memberikan nilai *kernel* pada baris 51 sampai 58 dan juga nilai  $\alpha_i$  pada baris 42 sampai 49 dari tiap data latih yang masuk dan diolah terlebih dahulu ditunjukkan pada baris 1 sampai 40.

#### 5.2.2.5 Algoritma Perhitungan Nilai $E_i$

Proses perhitungan nilai  $E_i$  dihitung dengan cara menjumlahkan hasil dari nilai matriks *Hessian* dengan nilai  $\alpha_i$  yang sudah di inisialisasi sebelumnya. Kode program perhitungan nilai  $E_i$  dapat dilihat pada table dibawah ini.

**Tabel 5.22 Proses Nilai  $E_i$**

Baris	Source Code
1	def getKelasData(self):
2	for i in self.data:
3	self.classified.append(i[-1])
4	# print(self.classified)
5	
6	def Hessian(self):
7	self.getKelasData()
8	result = []
9	for i in range(len(self.dataNormal)):
10	temp_result_in_line = []
11	for j in range(len(self.dataNormal)):
12	# print("pasangan ", j)
13	# print(self.dataNormal[i],
14	self.dataNormal[j])
15	temp_result_in_line.append(
16	
17	self.CalculateHessian(self.classified[i],
18	self.classified[j], self.dataNormal[i],
19	self.dataNormal[j]))
20	result.append(temp_result_in_line)
21	self.Hessian_Matrix = result
22	# print(self.classified)
23	# pprint.pprint(self.Hessian_Matrix, width=100)
24	
25	def calculateE(self, D, a):

26	$E = 0$
27	for $i$ in range(len(D)):
28	
29	$E += a[i]*D[i]*1.0$
30	return $E$

Tabel 5.18 diatas menunjukkan kode program yang menjelaskan perhitungan nilai  $E_i$  pada baris 25 sampai 30, dengan mengalikan nilai  $\alpha_i$  dengan nilai matriks Hessian yang sudah di dapat dan diproses pada baris 6 sampai 23.

### 5.2.2.6 Algoritma Perhitungan Nilai $\delta\alpha_i$

Perhitungan nilai  $\delta\alpha_i$  dengan cara mencari nilai minimal dari maksimal dari nilai gamma dikali dengan hasil nilai  $E_i$ , dikali dengan data ke- $a$ , dan nilai  $C$  dikurangi data ke- $a$ . kode program perhitungan nilai  $\delta\alpha_i$  dapat dilihat pada table dibawah ini.

**Tabel 5.23 Proses Nilai  $\delta\alpha_i$**

Baris	Source Code
1	def calculateDeltaAlpha(self, $E$ , $a$ ):
2	$\delta\alpha = \min(\max(\text{self.gamma}*(1.0-E), a),$
3	( $\text{self.constant}-a$ ))
4	return $\delta\alpha$

Tabel 5.19 diatas menunjukkan kode program yang menjelaskan perhitungan nilai deltaalfa ke- $a$  yang ditunjukkan pada baris 1 sampai 4 dengan perulangan terhadap nilai deltaalfa ke- $a$  sebanyak panjang data.

### 5.2.2.7 Algoritma Perhitungan Nilai $\alpha_i$

Perhitungan nilai  $\alpha_i$  dapat dihitung dengan cara menjumlahkan nilai  $\delta\alpha_i$  yang sudah didapat pada proses sebelumnya dengan nilai  $\alpha_i$  yang sudah di inialisasi yaitu bernilai 0. Kode program perhitungan nilai  $\alpha_i$  dapat dilihat pada table dibawah ini.

**Tabel 5.24 Proses Nilai  $\alpha_i$**

Baris	Source Code
1	def generateAlpha(self):
2	$\text{self.alphaList} = []$
3	$\text{self.deltaAlpha} = []$
4	$\text{self.Error} = []$
5	for $i$ in range(len( $\text{self.dataNormal}$ )):
6	$\text{self.alphaList.append}(\text{self.alpha})$
7	$\text{self.deltaAlpha.append}(1)$
8	$\text{self.Error.append}(0)$
9	def updateAlpha(self, $\delta\alpha$ , $a$ ):
10	return $a+\delta\alpha$

Tabel 5.20 diatas menunjukkan kode program yang menjelaskan perhitungan nilai  $\alpha$  yaitu dengan menjumlahkan nilai  $\alpha$  sebelumnya dengan nilai  $\delta\alpha$  yang sudah dihitung pada tahap sebelumnya, sesuai pada baris 1 sampai 10.

### 5.2.2.8 Algoritma Perhitungan Nilai $Kernelx^+$ , $Kernelx^-$ , dan $Bias$

Perhitungan nilai  $bias$  terlebih dahulu menghitung nilai  $kernelx^+$  dan  $kernelx^-$ , setelah mendapatkan nilai tersebut langkah selanjutnya menghitung nilai  $bias$ . Berikut merupakan kode program untuk mendapatkan nilai  $kernelx^+$ ,  $kernelx^-$  dan  $bias$  dapat dilihat pada table dibawah ini.

**Tabel 5.25 Proses Nilai  $Bias$**

Baris	Source Code
1	def calculateFX(self, indexPN):
2	hasil = 0
3	for i in range(len(self.alphaList)):
4	self.dataNormal[indexPN)
5	hasil += self.alphaList[i]*self.classified[i] * \
6	self.Hessian_Matrix[indexPN][i]
7	return hasil
8	
9	def calculateB(self):
10	hasil = 0
11	for i in self.indexPN:
12	hasil += self.calculateFX(i)
13	hasil = hasil*(-0.5)
14	# print("B", hasil)
15	return hasil

Tabel 5.21 diatas menunjukkan kode program yang menjelaskan perhitungan nilai  $kernelx^+$  dan  $kernelx^-$  yang ditunjukkan pada baris 1 sampai 7. Kemudian dilanjutkan perhitungan nilai bias yang ditunjukkan pada baris 9 sampai dengan baris ke 15.

### 5.2.2.9 Algoritma Perhitungan Nilai $Kerneltest$

Setelah mendapatkan nilai  $bias$ , langkah selanjutnya yaitu menghitung nilai  $kerneltest$  terhadap data latih dan data uji dengan menggunakan fungsi  $kernel$  sesuai dengan tahap *sequential training* yang sudah dihitung sebelumnya. Berikut kode program untuk menghitung nilai  $kerneltest$  dapat dilihat pada table dibawah ini.

**Tabel 5.26 Proses Nilai  $Kerneltest$**

Baris	Source Code
1	def generateMatrixKernel(self, xTest):
2	mat_ = []
3	for i in self.dataNormal:
4	mat_.append(self.CalculateKernel(i, xTest))
5	# print("matkernel", mat_)
6	return mat_

Tabel 5.22 diatas menunjukkan kode program yang menjelaskan perhitungan  $kerneltest$  dengan perulangan terhadap nilai  $kerneltest$  sebanyak panjang data yang ditunjukkan pada baris 1 sampai 6.

### 5.2.2.10 Algoritma Nilai Fxtest

Proses perhitungan nilai *fxtest* menggunakan data uji dan juga nilai *bias* yang sudah dihitung pada proses sebelumnya. Hasil nilai fungsi *linier* yaitu bernilai positif dan negatif. Jika hasil *fxtest* bernilai positif maka akan tergolong dalam kelas 1 yaitu kelas non-ikan dan jika bernilai negatif maka akan tergolong dalam kelas -1 yaitu kelas ikan. Kode program perhitungan nilai *fxtest* dapat dilihat pada table dibawah ini.

**Tabel 5.27 Proses Nilai Fxtest**

Baris	Source Code
1	<code>def fxTest(self, xTest):</code>
2	<code>    xTest = self.normalize(xTest)</code>
3	<code>    matrixKernel = self.generateMatrixKernel(xTest)</code>
4	<code>    hasil = 0</code>
5	<code>    # print("xtest", xTest)</code>
6	<code>    for i in range(len(self.alphaList)):</code>
7	<code>        hasil += (self.alphaList[i]*self.classified[i]</code>
8	<code>                * matrixKernel[i])</code>
9	<code>    hasil += self.calculateB()</code>
10	<code>    # print(hasil)</code>
11	<code>    return hasil</code>

Tabel 5.23 diatas menunjukkan kode program yang menjelaskan perhitungan nilai *fxtest* pada baris 1 sampai 11 dengan perulangan terhadap nilai *kerneltest* sebanyak data *kerneltest*. Kemudian menentukan hasil dari perhitungan *fxtest*.

### 5.2.2.11 Algoritma Tampilan Hasil Akhir pada LCD

Proses terakhir yaitu berupa tampilan dari hasil perhitungan pada python berupa penggunaan minyak masih layak atau tidak dengan melakukan pengiriman data *String* dengan menggunakan fungsi *Simplejson* dari *python* menuju ke Arduino IDE seperti yang ditunjukkan pada table 5.28.

**Tabel 5.28 Proses Menampilkan Hasil Pada LCD**

Baris	Source Code
1	<code>int sensorVal = digitalRead(Button);</code>
2	<code>while(Serial.available() &gt; 0){</code>
3	<code>    String hasil = Serial.readString();</code>
4	<code>    lcd.clear();</code>
5	<code>    lcd.setCursor(0, 0);</code>
6	<code>    lcd.print("Hasil Learning :");</code>
7	<code>    lcd.setCursor(0, 1);</code>
8	<code>    lcd.print(hasil);</code>
9	<code>    delay(10000);</code>
10	<code>}</code>
11	<code>if (sensorVal == 0 ) {</code>
12	<code>    String output;</code>
13	<code>    root.printTo(output);</code>
14	<code>    Serial.print(output+"\n");</code>
15	<code>    //Serial.println("Kirim Data");</code>
16	<code>    // while (1){</code>
17	<code>        lcd.clear();</code>
18	<code>        lcd.setCursor(0, 0);</code>
19	<code>        lcd.print("Kirim Data");</code>



## BAB 6 PENGUJIAN DAN ANALISIS

Bab ini membahas proses pengujian serta menganalisis hasil dari pengujian yang dilakukan berdasarkan sistem yang telah dibuat. Menunjukkan hasil proses pengujian dari tiap-tiap sensor dan komponen yang digunakan oleh peneliti. Kemudian melakukan pengujian system dengan menggunakan sebuah metode. Selanjutnya menghitung komputasi performa *system*. Adapun tujuan dilakukannya pengujian ada untuk mengetahui apakah semua kebutuhan yang diharapkan telah terpenuhi oleh sistem. Proses pengujian yang dilakukan yakni berupa pengujian fungsional, pengujian akurasi dan pengujian kecepatan pemrosesan sistem, dimana pengujian fungsional yakni menguji fungsi dari perangkat keras dalam hal ini berupa sensor-sensor yang digunakan serta LCD 16x2 apakah dapat bekerja spesifikasinya, pengujian akurasi yakni menguji seberapa akurat sistem yang telah dirancang dan diimplementasikan dibandingkan nilai atau hasil yang sebenarnya, sedangkan pengujian kecepatan pemrosesan sistem yakni untuk menguji waktu pemrosesan ketika sistem mulai di jalankan hingga menghasilkan jenis klasifikasi kelayakan minyak goreng dengan bahan ikan atau tahu. Berikut dijelaskan beberapa skenario pengujian yang dilakukan untuk menguji sistem.

### 6.1 Pengujian Fungsional Sistem

Pengujian fungsional merupakan pengujian yang berfungsi untuk mengetahui apakah keseluruhan sistem sudah berjalan sesuai dengan keinginan. Setiap komponen masukan dan keluaran akan diuji, diantaranya *Sensor Warna*, *Sensor Fotodiode*, dan LCD.

#### 6.1.1 Pengujian Sensor Warna TCS3200

Sensor warna TC3200 adalah sensor utama dalam sistem ini yang berfungsi untuk membaca nilai RGB dari warna minyak goreng yang dideteksi. Pada pengujian sensor warna ini akan dilakukan dengan melakukan pembacaan dari minyak goreng dengan penggorengan dengan bahan tertentu menggunakan sensor warna TCS3200 kemudian nilainya akan dibandingkan dengan alat ukur warna berupa *color PAINT*, dimana warna yang diuji menggunakan fitur *color* tersebut didapatkan dari pengambilan gambar minyak goreng yang diujikan.

##### 6.1.1.1 Tujuan

Tujuan dilakukannya pengujian ini adalah untuk mengetahui tingkat keakuratan sensor TCS3200 dalam membaca warna suatu objek. Dengan melakukan persamaan nilai antara yang dibaca oleh sensor dengan nilai warna yang terbaca di *color PAINT*, maka akan diperoleh selisih nilai error dari pembacaan sensor dan juga nilai R, G, B pada *color PAINT*.

### 6.1.1.2 Prosedur

1. Menghubungkan mikrokontroler Arduino Uno pada Laptop dengan menggunakan kabel port USB untuk melakukan proses komunikasi antara Arduino IDE dengan Arduino uno untuk melakukan proses akuisisi data.
2. Mengupload program untuk pembacaan warna ke Arduino Uno dari sensor TCS3200 dengan melakukan *mapping* terlebih dahulu pada *source code* sensor warna untuk mendapatkan hasil keakuratan sensor sesuai dengan nilai yang ditentukan dengan objek kalibrasi warna berupa kertas warna putih dan juga kertas warna hitam.
3. Mengukur nilai R, G, B pada 6 minyak goreng yang berbeda menggunakan sensor TCS3200 untuk mengetahui perubahan warna pada tiap minyak goreng yang diuji.
4. Menyesuaikan warna minyak goreng dengan mengambil gambar minyak goreng yang diujikan kemudian gambar tersebut diukur nilai R, G, B-nya menggunakan fitur *color PAINT*.
5. Menyamakan hasil pembacaan nilai R, G, B dari sensor warna yang ditampilkan pada LCD 16x2 dan mengamati nilai yang terbaca dengan nilai R, G, B dari gambar minyak goreng yang diukur dengan *color PAINT*, kemudian menentukan besarnya error pembacaan sensor dengan cara mengkonversi nilai R, G, B warna pembacaan sensor dan R, G, B warna dari gambar minyak goreng di *color PAINT* menjadi nilai HEX terlebih dahulu.
6. Nilai HEX dari warna masing-masing pembacaan warna kemudian diambil selisihnya untuk dilakukan perhitungan persentase error dalam bentuk nilai decimal.

Adapun cara untuk mengukur persentase *error* yaitu dengan menggunakan Persamaan (6.1) berikut :










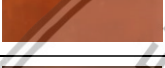





$$\text{Persentase error} = \frac{\text{Selisih Nilai Pembacaan}}{\text{Pembacaan Alat Ukur}} \times 100\% \quad (6.1)$$

Untuk menghitung nilai selisih pembacaan nilai sensor dan alat ukur yaitu dengan menggunakan Persamaan (6.2) berikut:

$$\text{Persentase error} = |\text{pembacaan alat ukur} - \text{pembacaan sensor}| \quad (6.2)$$

### 6.1.1.3 Hasil dan Analisis

**Tabel 6.1 Hasil pengujian sensor warna TCS 3200**

Peng ujian Ke-	Warna Minyak Goreng	Pembacaan Sensor TCS3200					Color tool Corel PHOTO-PAINT					Selisih Error		Error
		R	G	B	HEX	Warna	R	G	B	HEX	Warna	HEX	DEC	
1		146	119	132	#927784		148	112	29	#947a1d		#20299	131737	1,35 %
2		158	111	129	#9e6f81		116	118	19	#a67613		#80692	525970	4,82 %
3		189	109	128	#bd6d80		193	112	21	#c17015		#40295	262805	2,07 %
4		147	49	167	#9331a7		152	54	17	#983611		#5046a	328810	3,29 %
5		62	22	131	#3e1683		59	26	13	#3b1a0d		#2fc76	195702	5,05 %
Rata-rata														3,31 %

Berdasarkan Persamaan (6.1) untuk menentukan persentase error pada setiap pengujian, diperoleh rata-rata *error* sebesar 3,31 %. Contoh perhitungan persentase *error* pada pengujian ke-x sebagai berikut :

Nilai HEX warna pembacaan sensor = 9E6F81 (HEX) = 10383233 (DEC)

Nilai HEX warna pembacaan alat ukur = A67613 (HEX) = 10909203 (DEC)

$$\begin{aligned} \text{Persentase error} &= |\text{pembacaan alat ukur} - \text{pembacaan sensor}| \\ &= |10909203 - 10383233| \\ &= 525970 \end{aligned}$$

$$\begin{aligned} \text{Persentase error} &= \frac{\text{Selisih Nilai Pembacaan}}{\text{Pembacaan Alat Ukur}} \times 100\% \\ &= \frac{525970}{10909203} \times 100\% \\ &= 4,82\% \end{aligned}$$

Adapun untuk menghitung nilai rata-rata *error* keseluruhan pengujian adalah sebagai berikut:

$$\begin{aligned} \text{Rata - rata error} &= \frac{\text{Jumlah persentase error}}{\text{Jumlah pengujian}} \\ \text{Rata - rata error} &= \frac{1,35 + 4,82\% + 2,07\% + 3,29\% + 5,05\%}{5} \\ &= 3,31\% \end{aligned}$$

Rata-rata *error* yang dihasilkan sensor sangat kecil, sehingga dapat dikatakan bahwa akurasi pembacaan sensor warna sangat baik. Selain itu, walaupun terdapat *error* antara pembacaan sensor dengan alat ukur, namun sensor tetap dapat membaca warna meskipun perubahan nilainya cukup kecil, terlihat dari adanya perbedaan kepekatan warna yang ditunjukkan pada kolom warna *Color tool* Corel PHOTO-PAINT. Semakin gelap warna objek yang dideteksi maka nilai RGB semakin rendah (minimum 0,0,0,0), dan semakin terang warna yang dideteksi maka nilai RGB semakin tinggi (maksimum 255,255,225).

## 6.1.2 Pengujian Sensor Fotodiode

Sensor Fotodiode merupakan salah satu komponen utama dalam sistem ini yang berfungsi untuk membaca nilai Kekeruhan dari minyak goreng yang nantinya diolah menjadi Klasifikasi *Support Vector Machine*.

### 6.1.2.1 Tujuan

Tujuan dilakukan pengujian ini untuk mengetahui apakah Sensor Fotodiode sudah berjalan sesuai dengan keinginan peneliti. Pengambilan data untuk pengujian dilakukan sebanyak 12 kali lalu dibandingkan antara gambar jenis kekeruhan minyak dengan nilai pembacaan sensor yang ditampilkan di LCD16x2.




### 6.1.2.2 Prosedur

1. Menghubungkan mikrokontroler Arduino dengan Laptop.
2. Meng-*upload* kode program Sensor *Photodiode*.
3. Sistem diberikan jenis cairan yang memiliki kekeruhan berbeda-beda.
4. Mencatat nilai yang dihasilkan oleh Sensor Fotodiode ketika diberikan minyak goreng dengan tingkat kekeruhan yang berbeda.
5. Membandingkan nilai yang dihasilkan oleh pembacaan Sensor Fotodiode dengan parameter berupa gambar tingkat kekeruhan minyak.

### 6.1.2.3 Hasil dan Analisis

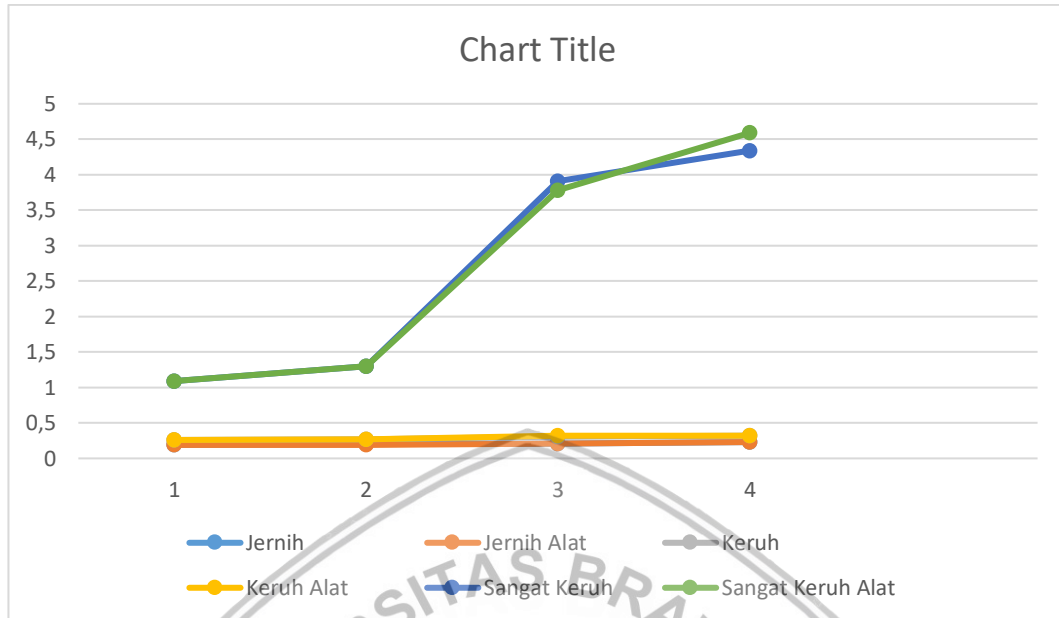
Dari 12 kali percobaan yang dilakukan, dapat dilihat hasil pengujian Sensor *Photodiode* yang ditampilkan pada *serial monitor*, lalu membandingkan nilai yang dihasilkan oleh sensor. Hasil pengujian dapat dilihat pada tabel 6.2.

**Tabel 6.2 Pengujian Sensor *Photodiode***

No.	Gambar Objek	Variabel Linguistik	Pembacaan Sensor (Didik, 2015) (volt)	Pembacaan Sensor Alat (volt)
1.		Jernih	0,19	0,19
			0,20	0,19
			0,21	0,21
			0,23	0,23
2.		Keruh	0,26	0,26
			0,27	0,27
			0,31	0,32
			0,32	0,32
3.		Sangat Keruh	1,09	1,09
			1,30	1,30
			3,91	3,78
			4,34	4,59



Gambar 6.1 Grafik Perbandingan Nilai kekeruhan



Hasil pengujian 12 kali yang telah dilakukan dapat dilihat bahwa sensor *photodiode* dapat membedakan kekeruhan dari sebuah cairan berdasarkan perbandingan dengan nilai pada jurnal yang sudah ada. Berdasarkan nilai sensor *photodiode* maka semakin keruh cairan akan semakin besar nilai *voltase*-nya. sebaliknya, semakin jernih cairan maka semakin kecil sensor membaca nilai *voltase* yang masuk. adapun pengujian kekeruhan berdasarkan fungsi kerja dari sensor dapat dilihat padan pada Table 6.2.

### 6.1.3 Pengujian Tampilan LCD

LCD 16x2 merupakan komponen untuk menampilkan karakter pada layar berukuran 16 kolom dan 2 baris. Karakter yang ditampilkan pada layar LCD pada umumnya berdasarkan apa yang telah ditentukan pada mikrokontroler, oleh karena ini pengujian ini dilakukan dengan melihat kesesuaian antara tampilan dengan rancangan yang sudah dibuat.

#### 6.1.3.1 Tujuan

Pengujian Tampilan LCD 16x2 ini bertujuan untuk mengetahui apakah hasil pembacaan nilai sensor dan hasil klasifikasi yang diharapkan pada saat ditampilkan pada layar LCD sesuai dengan kode program yang telah dirancang dan diimplementasikan.

#### 6.1.3.2 Prosedur




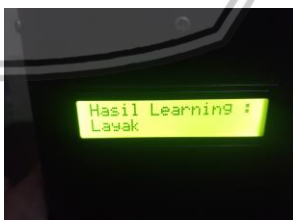

Untuk melakukan pengujian tampilan pada LCD 16x2 dilakukan dengan prosedur sebagai berikut :

1. Menghubungkan sistem kelayakan minyak goreng yang telah dibuat dengan laptop.
2. Mengupload kode program dari Sistem klasifikasi kelayakan minyak goreng.

- Mengamati tampilan pada LCD 16x2 dengan memberikan *Input* yang berbeda-beda saat melakukan pengujian untuk melihat perbedaan tampilan dari LCD 16x2 ketika adanya kondisi yang berbeda.

### 6.1.3.3 Hasil dan Analisis

**Tabel 6.3 Pengujian fungsional LCD 16x2**

No	Input Data	Output LCD	Status Nilai
1	<pre>lcd.print("R="); lcd.print("G="); lcd.print("B=");</pre>		Benar
2	<pre>lcd.print("Photodiode=");</pre>		Benar
3	<pre>lcd.print("Kirim Data");</pre>		Benar
4	<pre>lcd.print("Hasil Learning :"); lcd.print(hasil);</pre>		Benar
5	<pre>lcd.print("Hasil Learning :"); lcd.print(hasil);</pre>		Benar

Dari hasil pengujian dengan 5 kali masukan data, dapat dihitung nilai akurasi persentase tingkat kesalahan dari sistem yang dibuat. Rumus menghitung nilai persentase akurasi dapat dilihat pada persamaan (6.3).

Dengan hasil penghitungan sebagai berikut.

$$\text{Nilai Persentase Akurasi} = \frac{5}{5} \times 100\% = 100\% \quad (6.3)$$

Sistem ini memiliki nilai benar sebesar 100%, maka dapat disimpulkan sistem LCD dapat berjalan dengan lancar.

## 6.2 Pengujian Akurasi *Support Vector Machine*

Sistem klasifikasi penggunaan minyak goreng dengan menggunakan metode SVM ini mempunyai tujuan utama untuk dapat mengklasifikasikan tingkat kelayakan secara baik dan benar, oleh karena itu perlu diketahui tingkat keakuratan sistem dalam melakukan klasifikasi.

### 6.2.1 Tujuan

Tujuan dilakukannya pengujian ini adalah untuk menentukan nilai akurasi penggunaan metode *Support Vector Machine* pada sistem klasifikasi tingkat kelayakan minyak goreng yang telah dibuat.

### 6.2.2 Prosedur

Prosedur pengujian akurasi *Support Vector Machine* dilakukan adanya masukan 4 fitur berdasarkan pembacaan data sensor warna dan kekeruhan menggunakan rasio perbandingan antara data latih dan data uji yaitu 2:1. Penentuan proporsi data latih dan data uji dengan rasio 2:1 dikarenakan untuk mendapatkan akurasi semakin tinggi maka jumlah data latih harus lebih banyak minimal 2 kali lipat dari data uji (Adhieputra, 2010). Dimana dari jumlah data set sebanyak 120 data, 1/2 diantaranya dipilih secara acak untuk digunakan sebagai data latih yakni sebanyak 60 data dan 1/3 lainnya digunakan sebagai data uji yaitu sebanyak 13 data. Untuk menentukan nilai akurasi dari sistem yakni dengan membandingkan hasil klasifikasi penggunaan minyak goreng yang dilakukan oleh sistem dengan hasil klasifikasi kelayakan penggunaan minyak goreng dengan penghitungan manual. Rumus menghitung nilai persentase akurasi dapat dilihat pada Persamaan (6.4).

$$\text{Akurasi} = \frac{\text{Total data} - \text{Data tidak sesuai}}{\text{Total data}} \times 100\% \quad (6.4)$$

### 6.2.3 Hasil dan Analisis

Tabel 6.4 Pengujian akurasi system

No	R	G	B	P	Kelas	Hasil Sistem	Status
1	114	113	81	0.25	Layak	Layak	
2	50	45	40	0.28	Tidak Layak	Tidak Layak	
3	19	36	27	0.31	Tidak Layak	Tidak Layak	
4	79	72	44	0.27	Layak	Tidak Layak	
5	95	100	64	0.26	Layak	Layak	
6	1	29	34	1.23	Tidak Layak	Tidak Layak	
7	105	110	77	0.25	Layak	Layak	
8	3	28	26	1.08	Tidak Layak	Tidak Layak	
9	50	46	40	0.28	Tidak Layak	Tidak Layak	
10	84	92	61	0.26	Layak	Layak	
11	84	80	59	0.27	Layak	Layak	
12	15	20	22	0.29	Tidak Layak	Tidak Layak	
13	117	114	85	0.25	Layak	Layak	
Jumlah Nilai Benar							12

Berdasarkan hasil pengujian yang ditunjukkan pada Tabel 6.4 terlihat bahwa dari jumlah 13 data terdapat 1 hasil dari sistem yang tidak sesuai dengan kelas sebenarnya. Sehingga akurasi yang diperoleh Sistem klasifikasi penggunaan minyak goreng dengan Metode *Support Vector Machine* adalah sebesar 100%. Proses perhitungan akurasi sebagai berikut:

$$\text{Nilai Persentase Akurasi} = \frac{13 - 1}{13} \times 100\% = 92,3\%$$

Dari Hasil pengujian yang telah dilakukan didapat akurasi penghitungan nilai persentase sebesar 92,3%.

### 6.3 Pengujian Waktu Komputasi Sistem

Pada bab ini menjelaskan tentang proses komputasi waktu *learning machine* pada *python* untuk mendapatkan satuan waktu berupa *milisecon* yang nantinya akan menjadi penentu performa dari system yang sedang dijalankan.

#### 6.3.1 Tujuan

Tujuan dilakukannya pengujian ini untuk mengetahui waktu yang dibutuhkan sistem untuk melakukan proses pengklasifikasian jenis kelayakan

minyak goreng menggunakan metode *Support Vector Machine*, hal ini diperlukan untuk mengetahui performansi dari sistem yang telah dibuat.

### 6.3.2 Prosedur

Untuk melakukan pengujian waktu komputasi pemrosesan sistem dilakukan dengan cara mengukur waktu komputasi ketika program dimulai dan waktu komputasi ketika program selesai dalam satu siklus. Jika telah diketahui waktu komputasi awal dan akhir maka waktu yang dibutuhkan untuk sistem bekerja adalah nilai waktu komputasi akhir dikurangi dengan waktu komputasi awal. Prosedur pengujian ini dilakukan dengan menerapkan fungsi *time* pada kode program *python* sistem klasifikasi penggunaan minyak goreng ini. Fungsi *time* ini menghitung waktu dalam *millisecond*. Implementasi fungsi *time* pada *python* ditunjukkan pada Tabel 6.4 berikut.

**Tabel 6.5 Kode program perhitungan *time* pada *Python***

Baris	Source Code
1	<code>for q in fileData:</code>
2	<code>    t0 = time.time()</code>
3	<code>    benar = 0</code>
4	<code>    tester = loader.loadData(q, 1)</code>
5	<code>    . . . . .</code>
6	<code>    . . . . .</code>
7	<code>    for i in range(len(hasil)):</code>
8	<code>        if hasil[i] &gt; val:</code>
9	<code>            t1 = time.time()</code>
10	<code>            indx = i</code>
11	<code>            val = hasil[i]</code>
12	<code>    . . . . .</code>
13	<code>    . . . . .</code>
14	<code>total = t1-t0</code>

Pada Baris ke-2 terdapat inisialisasi variabel yang digunakan untuk penghitungan waktu awal atau *t0*. Pada baris ke-9 merupakan time akhir dari proses yang sedang berjalan atau *t1*, waktu akan berjalan terus menerus bertambah sesuai dengan panjang data latih, pada baris ke-14, memanfaatkan fungsi total *time* pada *python*, untuk mendapatkan waktu komputasi sistem yang diinginkan.

### 6.3.3 Hasil dan Analisis

**Tabel 6.6 Pengujian performa system**

No	Pengujian	Waktu (ms)
1	Pengujian ke-1	4496
2	Pengujian ke-2	4596
3	Pengujian ke-3	4567
4	Pengujian ke-4	4510



5	Pengujian ke-5	4391
6	Pengujian ke-6	4322
7	Pengujian ke-7	4221
8	Pengujian ke-8	4198
9	Pengujian ke-9	4556
10	Pengujian ke-10	4205
11	Pengujian ke-11	4219
12	Pengujian ke-12	4322
13	Pengujian ke-13	4383
Total		56.986

Berdasarkan hasil pengujian yang dilakukan sebanyak 13 kali, waktu komputasi sistem untuk melakukan pengambilan keputusan kelayakan minyak goreng rata-rata waktu sebesar 4384,53 ms atau sekitar 4,38 detik untuk melakukan penghitungan klasifikasi pengecekan penggunaan minyak goreng.

Berdasarkan hasil pengujian yang dilakukan sebanyak 13 kali, waktu komputasi sistem untuk melakukan klasifikasi jenis kelayakan minyak goreng dapat dihitung rata-ratanya menggunakan rumus.

$$\text{Rata - Rata} = \frac{56986}{13} = 4384,53 \text{ ms}$$

Dari hasil penghitungan rata-rata dapat disimpulkan sistem membutuhkan waktu sekitar 4384,53 ms untuk melakukan satu kali state proses penghitungan klasifikasi penggunaan minyak goreng.

## BAB 7 PENUTUP

### 7.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, pengujian dan analisis pada rumusan masalah yang ada, maka dapat diambil kesimpulan sebagai berikut:

1. Dalam penelitian ini telah berhasil dilakukan untuk pembuatan system implementasi metode klasifikasi *SVM* terhadap pemakaian minyak goreng dengan tingkat akurasi yang baik karena berdasarkan fungsi kerja *sensor photodiode* dapat membedakan tingkat kekeruhan minyak goreng dengan menaruh sensor *photodiode* dibawah gelas uji menghadap tepat didepan sensor warna dalam pengujian fungsional. Kemudian sensor TCS3200 ditempatkan diatas gelas uji dengan jarak pembacaan sensor kurang lebih sekitar 3cm dengan rata-rata error yang kecil sebesar 3,31%, sehingga dapat dikatakan sensor dapat bekerja dengan baik karena sudah dapat membaca nilai perubahan warna pada pengujian fungsional.
2. Dalam penerapan metode *SVM* pada system klasifikasi minyak goreng, didapatkan hasil learning dari komputasi nilai array R, G, B pada *TCS3200* dan nilai array P pada *photodiode* dapat berjalan sesuai dengan yang diinginkan. Dengan tingkat akurasi yang diperoleh dari system senilai 92,3 %. Dari jumlah keseluruhan data latih sebanyak 120 data dan data uji sebanyak 13 data. Sehingga dapat disimpulkan bahwa system ini memiliki tingkat akurasi yang baik dalam mengklasifikasi penggunaan minyak goreng pada pengujian akurasi data.
3. Pada pengujian komputasi waktu system Implementasi Metode Klasifikasi Support Vector Machine (*SVM*) Terhadap Pemakaian Minyak Goreng, didapatkan hasil waktu pemrosesan rata-rata sebesar 4384,53 ms dari 13 kali pengujian.

### 7.2 Saran

Berdasarkan hasil penelitian Implementasi Metode Klasifikasi Support Vector Machine (*SVM*) Terhadap Pemakaian Minyak Goreng, terdapat beberapa saran agar system ini dapat dikembangkan lebih lanjut.

1. Menggunakan algoritma *support vector machine* dengan menggunakan fungsi *kernel* yang lain untuk kemudian dapat diketahui perbandingan nilai hasil komputasi dari tiap *kernel* yang ada dan menggunakan fungsi klasifikasi *SVM* yang bisa membedakan lebih dari 2 kelas.
2. Melakukan kalibrasi sensor dengan menggunakan object berupa kertas berwarna sehingga nilai tingkat akurasi mapping dari sensor dapat bekerja lebih baik lagi dengan tingkat nilai *error* pembacaan sensor yang semakin kecil.
3. Menerapkan system yang sama kemudian ditampilkan pada *mobile apps* maupun *realtime* menggunakan *web* sehingga lebih memudahkan mobilitas pengguna.

4. System yang dibuat dapat membaca berbagai jenis minyak goreng yang ada di pasar maupun swalayan, sehingga tingkat keberhasilan dalam berbagai pengujian dapat menghasilkan nilai yang lebih baik dan akurat.



## DAFTAR PUSTAKA

- Adhieputra, D. A., 2010. *Pemilihan Data Training untuk Meningkatkan Kinerja Voting Feature Interval 5 (VFI 5)*. Bogor: SKRIPSI, Institut Pertanian Bogor, Ilmu.
- Aminah, S., 2010. Praktek Penggorengan Dan Mutu Minyak Goreng Sisa Pada Rumah Tangga Di Rt V Rw III Kedungmundu Tembalang Semarang. *Prosiding Seminar Nasional UNIMUS 2010*, Issue 6, pp. 261-267.
- AminahS, S., 2010. Bilangan Peroksida Minyak Goreng Curah Dan Sifat Organoleptik Tempe Pada Pengulangan Penggorengan. *Jurnal Pangan dan Gizi*, I(8), pp. 1-8.
- Arifin, B., 2015. Rancang Bangun Sistem Deteksi Minyak Goreng Yang Telah Dipakai Menggoreng Daging Babi Menggunakan LED dan Photodiode. Issue 5, pp. 1-6.
- Banzi, M., 2005. *Arduino*. [Online] Available at: <https://store.arduino.cc/usa/arduino-uno-rev3> [Accessed 6 8 2018].
- BSNI, 2013. *www.bsn.go.id*. [Online] Available at: <https://dokumen.tips/documents/sni-3741-2013-minyak-goreng-558463df91cf6.html> [Accessed 7 Januari 2018].
- Didik, E., 2015. Sistem Monitoring Digital Penggunaan dan Kualitas Kekeruhan Air PDAM Berbasis Mikrokontroler Menggunakan Sensor Fotodiode. *Jurnal Teknologi dan Sistem Komputer*, 3(e-ISSN: 2338-0403), p. 8.
- Jacobus, A., 2013. Penerapan Metode Support Vector Machine pada Sistem Deteksi Intrusi secara Real-time. *Berkala MIPA*, II(23), pp. 4-6.
- Malkan, I., 2015. Kualitas Minyak Goreng dan Produk Gorengan Selama Penggorengan di. *Jurnal Aplikasi Teknologi Pangan*, IV(2), pp. 1-5.
- Marofi, M. N., 2017. Rancang Bangun Sistem Klasifikasi Frekuensi Penggunaan Minyak Goreng dengan Menggunakan Metode Bayes. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, I(11), pp. 1-9.
- Nasional, B. S., 2013. Minyak Goreng. *Standar Nasional Indonesia (SNI) Minyak Goreng*, SNI 3741(22), pp. 1-23.
- Nora, S., 2016. Uji Kelayakan Minyak Goreng Curah dan Kemasan yang Digunakan Menggoreng Secara Berulang. *Jurnal Pendidikan Kimia*, 8(13), pp. 45-57.
- Prasetyawan, E. A., 2007. Uji Kualitas Minyak Goreng Pada Para Penjual Gorengan Di Lingkungan Kampus Universitas Jember. *Undergraduate Theses*, Issue 5, pp. 1-5.

- Putri, L. S., 2018. Klasifikasi Minyak Goreng Berdasarkan Frekuensi Penggorengan Menggunakan Metode K-Nearest Neighbor Berbasis Raspberry Pi. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 3(2), pp. 1-7.
- Rohmah, E. Z., 2017. Rancang Bangun Mesin Pertamina Digital Untuk Minyak Goreng. *Tugas Akhir Thesis*, Issue 1, p. 1.
- Satriyo, A., 2003. Support Vector Machine. *Teori dan Aplikasinya dalam Bioinformatika*, Volume I, pp. 1-11.
- Yang, T., 2015. Sistem Penyeleksi dan Pengelompokan Produk Berdasarkan Warna Berbasis PLC. *JURNAL ELEKTRO*, 8(13), pp. 31-44.
- Yuliana, Y., 2014. <http://digilib.unila.ac.id>. [Online] Available at: <http://digilib.unila.ac.id/92/> [Accessed 28 January 2018].

